CONTRIBUTIONS TO AUTOMATIC MULTIPLE F0 DETECTION IN POLYPHONIC MUSIC SIGNALS

Luís Felipe Velloso de Carvalho

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia Elétrica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Elétrica.

Orientador: Luiz Wagner Pereira Biscainho

Rio de Janeiro
Março de 2018

CONTRIBUTIONS TO AUTOMATIC MULTIPLE F0 DETECTION IN POLYPHONIC MUSIC SIGNALS

Luís Felipe Velloso de Carvalho

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Examinada por:

Prof. Luiz Wagner Pereira Biscainho, D.Sc.

Prof. Wallace Alves Martins, D.Sc.

Prof. Diego Barreto Haddad, D.Sc.

RIO DE JANEIRO, RJ – BRASIL
MARÇO DE 2018

Carvalho, Luís Felipe Velloso de

Contributions to automatic multiple F0 detection in polyphonic music signals/Luís Felipe Velloso de Carvalho. – Rio de Janeiro: UFRJ/COPPE, 2018.

X, 76 p.: il.; 29, 7cm.

Orientador: Luiz Wagner Pereira Biscainho

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia Elétrica, 2018.

Referências Bibliográficas: p. 67 – 74.

1. Multiple Pitch Estimation. 2. Automatic Music Transcription. 3. Musical Information Retrieval. I. Biscainho, Luiz Wagner Pereira. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

Agradecimentos

À minha família, pela paciência e carinho durante este tempo todo. Ao meu orientador Luiz Wagner Biscainho, pela oportunidade de ter realizado este trabalho em um tema ao mesmo tempo desafiador e interessante. Aos meus amigos, pelos inúmeros momentos de descontracão e pela companhia durante estes anos de mestrado.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

CONTRIBUIÇÕES PARA A DETECÇÃO AUTOMÁTICA DE MÚLTIPLAS F0S EM SINAIS DE MÚSICA POLIFÔNICOS

Luís Felipe Velloso de Carvalho

Março/2018

Orientador: Luiz Wagner Pereira Biscainho

Programa: Engenharia Elétrica

Estimação de múltiplas frequências fundamentais (MPE, do ingles *multi-pitch estimation*) é um problema importante na área de transcrição musical automática (TMA) e em muitas outras tarefas relacionadas a processamento de áudio. Aplicações de TMA são diversas, desde classificação de gêneros musicais ao aprendizado automático de piano, as quais consistem em uma parcela significativa de tarefas de extração de informação musical. Métodos atuais de TMA ainda possuem um desempenho consideravelmente ruim quando comparados aos de profissionais da área, e há um consenso que o desenvolvimento de um sistema automatizado para a transcrição completa de música polifônica independentemente de sua complexidade ainda é um problema em aberto.

O objetivo deste trabalho é propor contribuições para a detecção automática de múltiplas frequências fundamentais em sinais de música polifônica. Um método de referência para MPE é primeiramente escolhido para ser estudado e implementado, e uma modificação é proposta para melhorar o desempenho do sistema. Por fim, três estratégias de refinamento são propostas para serem incorporadas ao método modificado, com o objetivo de aumentar a qualidade dos resultados. Testes experimentais mostram que tais refinamentos melhoram em média o desempenho do sistema, embora cada um atue de uma maneira diferente de acordo com a natureza dos sinais.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

CONTRIBUTIONS TO AUTOMATIC MULTIPLE F0 DETECTION IN POLYPHONIC MUSIC SIGNALS

Luís Felipe Velloso de Carvalho

March/2018

Advisor: Luiz Wagner Pereira Biscainho

Department: Electrical Engineering

Multiple fundamental frequency estimation, or multi-pitch estimation (MPE), is a key problem in automatic music transcription (AMT) and many other related audio processing tasks. Applications of AMT are numerous, ranging from musical genre classification to automatic piano tutoring, and these form a significant part of musical information retrieval tasks. Current AMT systems still perform considerably below human experts, and there is a consensus that the development of an automated system for full transcription of polyphonic music regardless of its complexity is still an open problem.

The goal of this work is to propose contributions for the automatic detection of multiple fundamental frequencies in polyphonic music signals. A reference MPE method is chosen to be studied and implemented, and a modification is proposed to improve the performance of the system. Lastly, three refinement strategies are proposed to be incorporated into the modified method, in order to increase the quality of the results. Experimental tests reveal that such refinements improve the overall performance of the system, even if each one performs differently according to signal characteristics.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The art of combining melodic and rhythmic sounds has always fascinated and touched the human being, and music can be considered an omnipresent and essential part of our lives. One of the most important and influential forms of cultural manifestation, music is a relevant cultural attribute for characterising aspects of the society in the present and in the past.

It is argued by many that one of the most valuable achievements of humankind in the context of music was the ability of transcribing a piece of music into a human readable and interpretable representation, so that it can be performed by another or the same musician afterwards. Ancient civilisations were probably aware of the importance of this task, with the oldest known music notation dating back to approximately 1400 BC [1].

Musical information retrieval (MIR) is an emerging multidisciplinary field of research that aims to extract meaningful content from music data [2] by exploiting concepts from several areas, such as signal processing, machine learning, music theory, and psychology. One of the most important and challenging tasks of MIR is the *automatic music transcription* (AMT), which is the process of converting a music signal into some kind of comprehensible symbolic representation with the help of computer processing. It is important because it has numerous applications in multiple levels, covering topics from interactions with music to audio coding, as described in Section 1.1; and it is considered significantly challenging because AMT comprises

several complex individual subtasks, including instrument recognition, extraction of rhythmic information, multi-pitch estimation, note onset/offset detection, source separation, among others, which are still active research topics. An overview of AMT approaches can be found in [3].

The core problem in AMT is to estimate concurrent pitches in a time frame[1] [4] (the case of *polyphonic* music, that is, when two or more sounds occur simultaneously), and this task is referred to as *multi-pitch detection* (MPE) or *multiple fundamental frequency estimation* (MF0E). The common AMT procedure that follows MPE is *note tracking*, which is the estimation of continuous segments that usually correspond to individual notes [5]. If fact, the great majority of AMT research addresses only MPE and note tracking (which can be done either jointly or separately) [6].

A music transcription example is depicted in Figure 1.1. The waveform input is a recording of W.A. Mozart's trio for clarinet, viola and piano in E♭ major – "Kegelstatt Trio" (K.498), and the output is a score with the three instruments. It should be mentioned that, for a complete transcription as shown in Figure 1.1, all AMT subtasks must be carried out. It can be seen that the score of each instrument is indicated separately (instrument identification and separation tasks) and the metrical structure is estimated (extraction of rhythmic information). Performing MPE followed by note tracking is considered only partial transcription, and its results can be expressed by means of a piano roll representation (see Subsection 2.1.2).

1.1 Applications

The goal of this section is to motivate research on multiple fundamental frequency estimation by highlighting some applications that benefit from it, either direct or indirectly. For a more detailed discussion of applications of MPE, the reader may refer to [3, 7, 8].

As mentioned before, the main application of MPE is in developing systems for

[1] A short time interval; this concept is defined in Chapter 3.

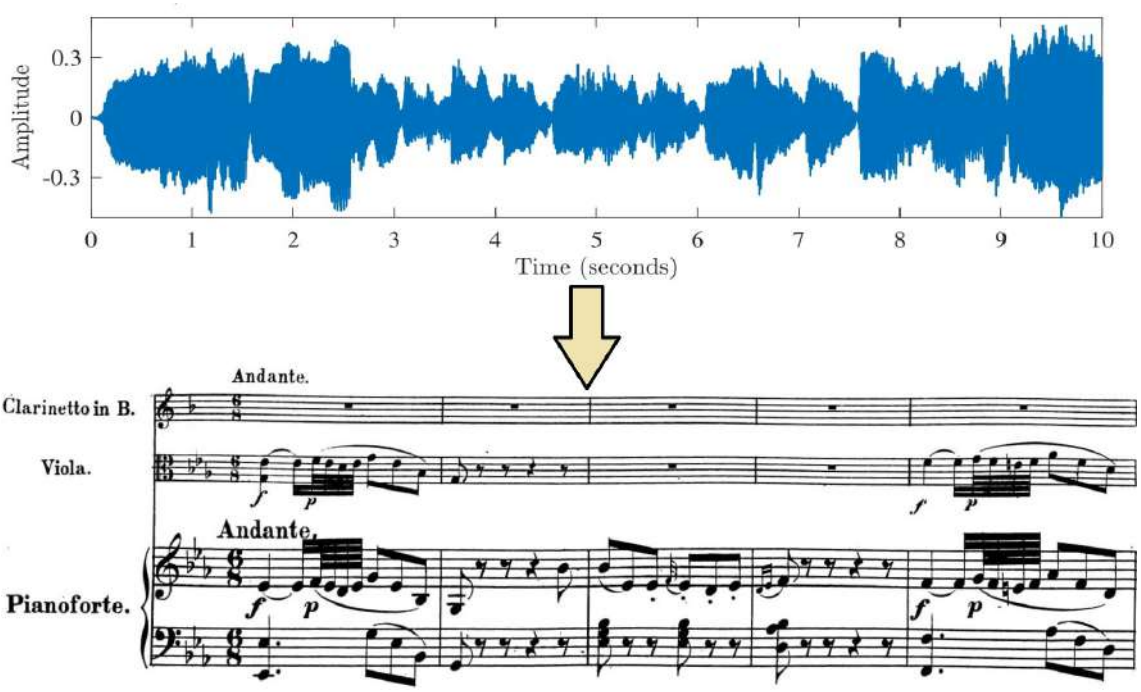

Figure 1.1: A music transcription example of W.A. Mozart's trio for clarinet, viola and piano in E♭ major – "Kegelstatt Trio" (K.498). Top plot represents the input waveform, and bottom figure indicates the output score.

the automatic transcription of music. Even for trained musicians, the transcription of complex polyphonic music is a difficult task, and current AMT systems still perform considerably below human experts [6]. There is a consensus that the development of an automated system for full transcription of polyphonic music regardless of its complexity is still an open problem [4].

The range of AMT applications is significantly broad, since this is a key problem in MIR. Probably the most straightforward application of AMT is to help a musician to transcribe a recorded performance. The musician may also benefit from automatic transcription tools for composition assistance, since a musical score also permits making musically significant modifications to the music rather than just storing it [7]. Still in [7], Klapuri says that AMT tools can promote an active attitude for professional and amateur musicians by stating that "the relaxing effect of the sensomotoric exercise of performing and varying good music is quite a different thing than merely passively listening to a piece of music".

From the musicology perspective, AMT systems can be useful to facilitate the

analysis of music genres whose scores are not available, such as music from oral traditions [9] or improvised music, like jazz [10]. Other MIR tasks that can benefit from AMT tools include genre classification [11], plagiarism detection [12], and query-by-humming [13], among others.

The constant increase in the computational capacity of modern processors also allows the development of real-time AMT systems. Applications of online transcription include the creation of interactive platforms for musical instrument tutoring, such as Yousician[2], where the learner's performance can be automatically evaluated based on a reference score [14]. Real-time AMT can be also used to develop automatic music accompaniment systems, which are systems able to follow human performances (either monophonic or polyphonic) by adjusting themselves accordingly [15].

Although much of MPE research is applied to automatic music transcription, this task is also crucial in other sub-areas of audio processing. The emerging field of sound scene analysis [16], which refers to the development of systems for automatic detection and classification of everyday sound scenes, benefits from MPE, since diverse acoustic scenarios include lots of concurrent pitched sounds; sound scene analysis has important and potential applications such as urban planning, acoustic ecology, smart homes, and audio-based surveillance. MPE is also applied in speech processing, for instance in the extraction of intonation patterns for speech recognition [17].

1.2 Goals

The goal of this work is to propose contributions to the automatic detection of multiple fundamental frequencies in polyphonic music signals. For practical purposes, this goal can be divided into subtasks, which are listed as follows:

- Study and implementation of a reference method for multi-pitch detection;

[2] `https://yousician.com/`

- Propose modifications for this method in order to improve its performance;
- Evaluate the proposed methodology on reference datasets, so that results can be compared with those produced by methods from the literature;
- Implement refinements that can be integrated into the system in order to improve the quality of the results.

1.3 Organisation of the work

This work is organised as follows. In Chapter 2, background concepts are presented and a literature review is carried out. It begins in Section 2.1 by introducing basic concepts about music signals and terminology. After that, reference methods for automatic music transcription and multiple fundamental frequency detection are described in Section 2.2.

In Chapter 3, the main method for multiple fundamental frequency estimation approached in this work is described. In Section 3.1, a motivation for the methodology is presented, as well as an overview of the system. Following that, the method and its modification are described in Section 3.2. To conclude the chapter, the performance of the method is assessed in Section 3.3.

In Chapter 4, three refining algorithms are proposed to be integrated into the methods. First, a user interaction approach is proposed in Section 4.1, followed by two post-processing refinements, described respectively in Sections 4.2 and 4.3. Lastly, in Chapter 5, conclusion and future directions of this work are drawn.

Chapter 2

Background and literature review

2.1 Basic concepts

This section is dedicated to the definition and clarification of important concepts about musical signals, as well as fundamental elements of automatic music transcription and multi-pitch estimation that will be essential for the understanding of the following chapters.

2.1.1 Characteristics of music signals

If someone were asked to enunciate a definition for music, it is very likely that the answer would include words like "emotion", "pleasure", or "beauty". In fact, music is defined by the Oxford English Dictionary[1] as "vocal or instrumental sounds (or both) combined in such a way as to produce beauty of form, harmony, and expression of emotion". In no way this is a precise definition, and since the present work deals with music signals, a more technical definition should be adopted, unfortunately excluding those subjective parts evoked in listeners.

Music signals are the subset of audio signals that are a combination of musical notes from one or more musical instruments, including singing voice. Moreover, since the majority of automatic music transcription systems are designed and implemented

[1] https://en.oxforddictionaries.com/

in computers (including this work), the signals to be analysed should be available in a digital format [18].

Fundamental frequency, harmonicity and pitch

A signal is defined as periodic when it repeats at regular time intervals, whose shortest duration is referred to as the *fundamental period* [19]; the inverse of the period is defined as the *fundamental frequency* $F0$ of the signal. Musical instruments are usually constructed to produce sounds with controlled and stable fundamental periods [20], which can be modelled as signals composed of a combination of sinusoids at integer multiples of a fundamental frequency. These components are referred to as *harmonics*.

However, for some instruments the sinusoidal components are not placed at exactly integer multiples of the fundamental frequency (*e.g.* stiff string instruments), and these instruments are said to present inharmonic components. Therefore, the concept of *partial* can be conveniently introduced to comprise both harmonic and inharmonic components.

The partials of a harmonic instrument sound produce the perception of a clearly defined *pitch*, which is a concept closely related to the fundamental frequency [19]. Klapuri ([3], Chapter 1) defines pitch as "a perceptual attribute which allows the ordering of sounds on a frequency-related scale extending from low to high", while Hartmann [21] says "that a sound has a certain pitch if it can be reliably matched by adjusting the frequency of a sine wave of arbitrary amplitude". In this work, the terms pitch (which is usually reserved for perceived frequency) and fundamental frequency are used interchangeably.

Musical notes

A *musical note* can be defined as a single sound event produced by a pitched instrument [22], and can be described by a collection of attributes. The starting time of the note is referred to as its *onset*, and its temporal end is the *offset* [23]. The note

duration is intuitively defined as the difference between these time instants. Also, each note is assigned to a specific perceived pitch, one of its most distinguishing attributes. It is important to mention that pitch does not always remain constant within a note, since it is quite common the presence of modulation effects (*e.g.* vibrato) performed by the musician.

In order to facilitate music description, it is convenient to quantise the space of all possible pitches, leading to the notion of a musical scale [22]. Western music is usually described by the twelve-tone equally tempered scale, consisting of twelve pitch classes represented by the letters C, D, E, F, G, A and B, and the accidents sharp (♯) and flat (♭). The distance between two consecutive notes in this scale is called a *semitone*, and a *tone* equals two semitones.

One of the most commonly adopted naming standards for musical notes is the Scientific Pitch Notation [24], where each note is specified by the pitch class name, followed by a number representing its octave. In this notation, for instance, D5 represents the note D from the fifth octave, and E♭3 refers to the note E-flat played at the third octave. The reference note is the A4, whose fundamental frequency is standardised to 440 Hz.

Lastly, it is important to define the concept of *timbre*, which is the attribute that distinguishes sounds produced by different musical instruments. For example, this is the attribute that allows the listener to distinguish the same musical note as played by a trumpet or by a violin. Timbre is a perceptual property of sound, and many attempts have been made to characterise it in terms of objective measures such as temporal and spectral evolution, or the energy distribution across the partials of a fundamental frequency associated with a musical note [22].

2.1.2 The MIDI protocol and its notation

The Musical Instrument Digital Interface (MIDI) protocol [25] is a very popular computer music notation employed to store musical scores and communicate information between digital music devices. The MIDI files are specified as a collection

of notes, each one carrying event messages such as its onset, duration, pitch and intensity. Also, in this standard each pitch is mapped into a real number q according to the following expression:

$$q(F0) = 69 + 12 \log_2 \left(\frac{F0}{440 \text{ Hz}} \right), \tag{2.1}$$

$F0$ being the fundamental frequency of the associated musical note.

MIDI data can be a reliable and powerful representation to some extent. It is a simple format that requires only a few kilobytes to store entire songs, a popular standard amongst both music professionals and researchers, and suffices for many MIR research questions. On the other hand, it does not store proper music notation and cannot handle events such as expressive features and instrument timbre information. A recent study on how this protocol can be exploited in MIR applications can be found in [26], including a Python toolbox for extracting information from MIDI files.

As mentioned before, MIDI files carry some of the most important attributes of the stored musical notes, in addition to the fact that they also support multiple channels. Such characteristics make MIDI files suitable to be employed in MPE and AMT tasks. When a music recording and its associated MIDI file are available and they are synchronised to each other, the MIDI file can be used as ground truth[2] for quality assessment of the transcription. When the files are not synchronised (*e.g.* in different interpretations of an original piece), the MIDI representation can be used in score-informed tasks, such as score-informed transcription [14] or source separation [27], and score following [28], which is the real-time alignment of an incoming music signal to its respective music score.

A typical graphical depiction of MIDI content is the piano-roll, a mid-level representation that resembles a spectrogram (see Subsection 3.2.1). Figure 2.1 shows an example of a piano-roll for an excerpt of J.S. Bach's Sarabande from Partita in A minor for Solo Flute (BWV 1013), generated with the MIDI Toolbox for Matlab [29].

[2]This term refers to the reference transcription to be compared with.

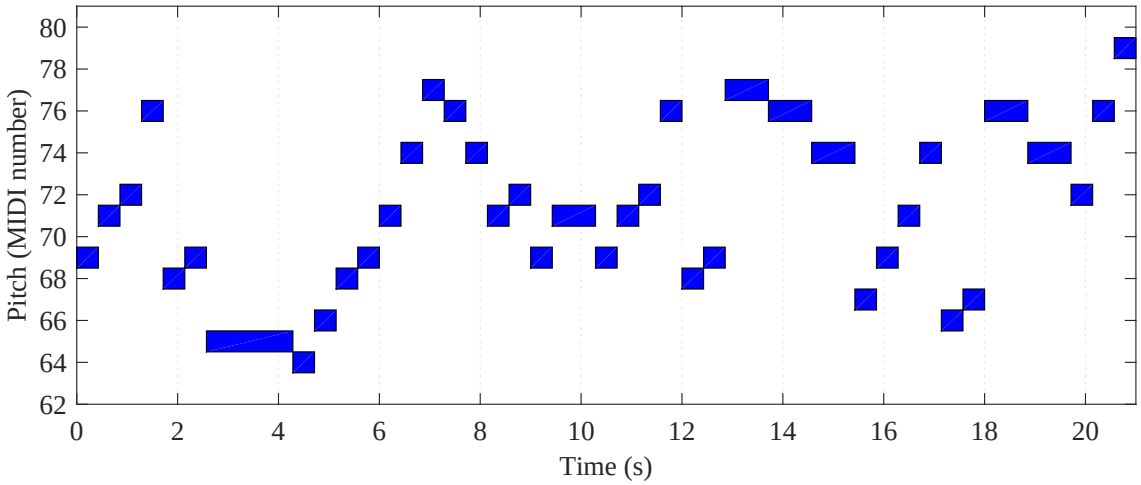

Figure 2.1: The piano-roll representation of an excerpt of J.S. Bach's Sarabande from Partita in A minor for Solo Flute (BWV 1013). The horizontal and vertical axes encode time and pitch number, respectively. Here, musical notes are described by three parameters: onset, duration and pitch number.

2.1.3 Datasets

The use of benchmark datasets is essential to ensure research sustainability. In this subsection, the collection of musical signals employed in this work are established.

Bach10 dataset

The Bach10 dataset [17] is a free[3] polyphonic music dataset, consisting of audio recordings of ten pieces of four-part J.S. Bach chorales (scored for violin, clarinet, saxophone, and bassoon). It also contains the non-synchronised respective MIDI files, as well as the ground truth pitch information of each part, therefore making the collection suited to multiple fundamental frequency estimation tasks.

Since the isolated recordings of each instrument for each piece are also available, it is possible to explore different instrumentations or polyphony levels (*i.e.* the number of concurrent sounds) by combining different parts of each piece, thus expanding the dataset up to 150 recordings: 40 solos, 60 duets, 40 trios, and the ten original quartets. For example, one may desire to evaluate an algorithm performance as the number of concurrent instruments increase.

This dataset has been employed in the past for evaluation of MPE tasks by Duan

[3]Available on `http://music.cs.northwestern.edu/data/Bach10.html`

et al. [17], Cheng *et al.* [30], Sigtia *et al.* [31], Benetos and Weyde [32], and Kraft and Zölzer [33, 34].

MIREX mulfiF0 development dataset

This multi-track recording is used as a development set for the MIREX[4] multi-$F0$ and note tracking tasks [35]. It consists of a woodwind quintet transcription of the fifth variation from L. van Beethoven's Variations for String Quartet Op.18 No. 5, accompanied with the ground truth pitch values. Like the Bach10 collection, this set also contains the isolated recording of each instrument (flute, oboe, clarinet, horn and bassoon), thus allowing the set to be expanded to five solos, ten duets, ten trios, five quartets, and the original quintet, yielding 31 annotated music signals.

This dataset has been employed in the past for evaluation of MPE tasks by Vincent *et al.* [36], Benetos and Dixon [37, 38], Carabias-Orti *et al.* [39], Grindlay and Ellis [40], Benetos and Weyde [32, 41], Cheng *et al.* [30], and Kraft and Zölzer [33, 34].

TRIOS

The TRIOS dataset [42] is a collection of five multi-track recordings of short musical extracts from trio pieces of classical and jazz music. Like the aforementioned datasets, this collection supplies the isolated recordings of each instrument (bassoon, cello, clarinet, drums, horn, piano, alto saxophone, trumpet, viola, and violin) with their respective aligned MIDI files. Therefore a 35-item collection can be generated, consisting of 15 solos, 15 duets, and the 5 original trios.

This dataset has been employed in the past for evaluation of MPE tasks by Benetos and Weyde [32, 41], Benetos *et al.* [43], and Kraft and Zölzer [33, 34].

[4] Annual event for public evaluation of MIR algorithms.

2.1.4 Evaluation metrics

In many multiple $F0$ estimation algorithms, the input signal is divided into analysis frames by means of a time-frequency representation (see Subsection 3.2.1), and $F0$ candidates are estimated in each frame. Therefore, since this framework is adopted here, frame-wise metrics are employed to compare the estimated pitches with the ground truth.

For evaluating and comparing MPE methods, the chosen metrics are the precision, recall and F-measure (defined in [44]), which are respectively expressed by:

$$\mathcal{P} = \frac{\sum_b N_{\mathrm{TP}}[b]}{\sum_b N_{\mathrm{TP}}[b] + \sum_b N_{\mathrm{FP}}[b]} \tag{2.2}$$

$$\mathcal{R} = \frac{\sum_b N_{\mathrm{TP}}[b]}{\sum_b N_{\mathrm{TP}}[b] + \sum_b N_{\mathrm{FN}}[b]} \tag{2.3}$$

$$\mathcal{F} = \frac{2 \cdot \mathcal{R} \cdot \mathcal{P}}{\mathcal{R} + \mathcal{P}}, \tag{2.4}$$

where precision is the ratio between true positives $N_{\mathrm{TP}}[b]$ (*i.e.* the number of correctly estimated pitches) and the sum of true positives and false positives $N_{\mathrm{FP}}[b]$ (*i.e.* the total number of detected pitches), recall is the ratio between true positives and the sum of true positives and false negatives $N_{\mathrm{FN}}[b]$ (*i.e.* the total number of ground-truth pitches), and F-measure is the harmonic mean of precision and recall. The term b refers to the frame index, and the above-mentioned quantities are summed across all frames before computation of the ratios.

Also, less popular than precision, recall, and F-measure but still sometimes found in literature, the accuracy (defined in [45]) is employed:

$$\mathcal{A} = \frac{\sum_b N_{\mathrm{TP}}[b]}{\sum_b N_{\mathrm{TP}}[b] + \sum_b N_{\mathrm{FP}}[b] + \sum_b N_{\mathrm{FN}}[b]}, \tag{2.5}$$

which is a measure of the overall performance.

For the four metrics aforespecified, a frequency estimate is labelled as correctly if it is within one semitone of the true fundamental frequency [44].

2.2 Literature review

The goal of this section is to review related studies on multi-pitch estimation. As mentioned in Chapter 1, many systems proposed for AMT address only the MPE task. Therefore, previous works on AMT are also reviewed within this section.

While multi-pitch estimation is still considered an open problem, the problem of single-pitch estimation to automatic transcription of monophonic music has been already solved [4, 6]. YIN [46] and its modification PYIN [47] figure as the best methods for AMT of monophonic signals.

For a more detailed overview on MPE and AMT methods, the reader may refer to [4, 6, 18, 19]. In [4, 18], MPE systems are categorised according to their core methodologies, and in [19] Yeh classifies them according to their estimation type whether iterative or joint. While joint estimation techniques evaluate a set of possible combinations of multiple pitch hypotheses, iterative methods estimate the most prominent pitch and perform suppression of related sources at each iteration, until a termination criterion is met. Yeh asserts that this categorisation is more convenient, since advantages and drawbacks of each one are opposed: joint methods can handle more efficiently interactions between concurrent sources, but require significant computational complexity; on the other hand, iterative techniques are computationally inexpensive, but their errors tend to accumulate at each iteration step, thus compromising their performance. Here, the iterative/joint approaches will be first explored, followed by a review of some methods according to their core methodologies.

As for iterative estimation methods, the work of Anssi Klapuri[5] is an important contribution to the field of automatic music transcription. In the method proposed in [48], the spectrum of the input signal is split over a 2/3 octave filterbank, and fundamental frequency weights are calculated for each band according to a salience function that allows for inharmonicity. After that, partials of the fundamental frequency with the highest global weight are smoothed and then subtracted from the mixture.

[5] `http://www.cs.tut.fi/~klap/iiro/index.html`

The process repeats until a criterion based on a polyphony inference method is met. In [49], Klapuri proposes an alternate technique that uses an auditory filterbank. The signal in each subband is processed, and the results are summed across channels to yield a summary spectrum. The most prominent fundamental frequency is obtained via a salience function, and then removed from the mixture by a cancelling algorithm. The process stops when a salience condition is met. In [50], an improved version of this method is proposed using a computationally efficient auditory model.

A wide range of joint MPE methods have been proposed in the literature. One of the techniques with best performances is the one proposed by Yeh in [19]. In his approach, sinusoidal components are first extracted from the input signal using an adaptive noise level estimation. After that, fundamental frequency candidates are selected to form a set of multiple $F0$ hypotheses. The hypothetical sequences are then evaluated by a score function that takes into account three physical principles: harmonicity, spectral smoothness, and synchronous amplitude evolution. Lastly, a polyphony inference algorithm selects candidates with the highest scores.

Many proposed methods address MPE via statistical spectral models. In [51], multi-pitch estimation of piano notes is viewed as a maximum *a posteriori* estimation problem, given a time instant and a set of all possible fundamental frequency combinations. Firstly, the spectral envelope of the overtones of each note is modelled with a smooth autoregressive model as a likelihood function. Then, the pitch candidates that maximises this likelihood function are selected from a set of possible pitch combinations. In [17], Duan *et al.* propose an MPE technique that uses a likelihood function to model the input signal spectrum in peak and non-peak regions, the former being the probability of a true salient peak being detected and the latter its complimentary version. Parameters of these models are learned from training data, and a polyphony estimation algorithm is additionally proposed.

The vast majority of MPE strategies exploit time-frequency representations derived from either spectral or temporal structures. However, there are a few related

works that benefit from spectrotemporal representations[6], which are temporal representations computed after splitting the input signal through a filterbank. Meddis and O'Mard proposed a unitary model of pitch perception [52], where the input signals (addressed by the authors as stimuli) are split through a 60-channel linear fourth-order gammatone filterbank to represent the mechanical frequency selectivity of the human cochlea. Band-wise individual autocorrelation functions are then computed and directly summed across the channels in order to produce a summary autocorrelation function (SACF), followed by a threshold-based judgement on the SACF to select strong pitch candidates. This model exhibits high complexity due to the considerable number of channels involved in its computation, and a few years later Tolonen and Karjalainen [53] proposed a simplification of the Meddis and O'Mard's model, where the SACF is calculated from splitting the input signal into only two channels, and further processed to yield an enhanced SACF (ESACF), from which potential concurrent fundamental frequencies are estimated. In 2014 Kraft and Zölzer [33] revisited the Tolonen and Karjalainen model and improved the peak selection method by carrying out an iterative analysis of the SACF.

A significant number of MPE/AMT methods have been proposed employing spectral factorisation algorithms. Non-negative matrix factorisation (NMF) and probabilistic latent component analysis (PLCA) figure as the main adopted techniques. An extensive review of spectrogram factorisation strategies for AMT is carried out by Benetos in his PhD thesis [4]. NMF is a method that decomposes a non-negative matrix as a product of two other matrices, also non-negative, using an optimisation algorithm that minimises the distance between the input matrix and the resulting product via a cost function. Since the magnitude or the power spectrogram of a music signal is non-negative by definition, a sound mixture can be modelled as a linear combination of the individual sources; therefore the NMF is a suitable method for decomposition of an input spectrogram in terms of non-negative elements, each one representing different parts of a single source. NMF was first ap-

[6]This class of representation is explored with more details in Chapter 3.

plied to polyphonic AMT by Smaragdis and Brown [54]. In [55], it was proposed an extension for the NMF of [54] that integrated sparseness constraints into the optimisation process. In [36], harmonicity constraints were incorporated into the NMF model, yielding two distinct, harmonic and inharmonic, NMF algorithms.

PLCA is an alternative probabilistic formulation of NMF, first introduced for AMT/MPE in [56]. In this technique, the input spectrogram is modelled as a probability distribution to be afterwards decomposed into a product of one-dimensional marginal distributions, also using an optimisation process that minimises a cost function. Grindlay and Ellis proposed in [40] an extension of the PLCA that supports multiple spectral templates for each pitch and instrument, with these templates being defined as *eigeninstruments*. In [57], Benetos and Dixon proposed a convolutive PLCA algorithm that supports frequency modulations, and also uses multiple spectral templates per pitch and instrument. In [38], Benetos and Dixon extended the convolutive PLCA of [57] by incorporating spectral templates that correspond to musical note states such as attack, sustain, and decay, whose order is regulated by means of hidden Markov model-based temporal constraints. In [43], this model is improved by integrating pre-extracted and pre-shifted musical note templates from multiple instruments, and also a scheme employing massive parallel computations with graphics processing units (GPUs) is proposed to yield faster transcriptions. In [32], Benetos and Weyde proposed an efficient extended version of [38] that increases the dimension of the note state templates, resulting in the elimination of the convolution stage; as a result, this model is considerably faster than its original version.

One drawback of spectral factorisation methods, such as NMF and PLCA, is that they tend to converge to a local minimum, since they are significantly sensitive to initialisation parameters. Cheng *et al.* [30] address this issue by proposing a PLCA technique that benefits from a deterministic annealing expectation-maximization (EM) algorithm in order to escape from local minima. This method modifies the PLCA algorithm of [57] by incorporating a "temperature" parameter into the opti-

misation rules, so that a more refined optimisation can be performed at each update step.

In [6, 58], a group of researchers from the Centre for Digital Music at Queen Mary University of London[7], one of the largest centres in the world for research in music technology, made a study on recent AMT methods and their respective performances, in order to discuss the current challenges and future directions of this area towards a complete transcription. It is argued that, although the AMT area is significantly active, the performance of recent systems seems to have reached a certain limit. Current transcription methods tend to employ general purpose models in their algorithms. However, the diversity of music signals is broad and, as a result, such systems fail in reliably transcribing music signals from different instruments and/or musical styles. The study concludes that a way to improve AMT systems is to tailor them for sub-cases, such as specific musical instruments or genres. This can be done by extracting models that reflect musical conventions about the piece in question, such as acoustic, statistical, or musicological models. Additionally, user interaction is also suggested as an approach to improve the quality of the transcription. For instance, the original score of a specific performance of a piece can be used as a prior information for the system, or the user could incorporate information regarding the rhythmic structure of the piece.

[7] http://c4dm.eecs.qmul.ac.uk/

Chapter 3

Multiple fundamental frequency estimation

In this chapter the core method for multiple fundamental frequency detection is described. It begins by motivating the chosen approach and also by drawing an overview of the method steps in Section 3.1. The method is then described in Section 3.2, including a simple solution for dealing with non-integer MIDI numbers. Results for multi-pitch detection over the signals from the target databases (see Subsection 2.1.3) when using both the original and modified methods are presented and compared in Section 3.3. Lastly, final considerations are carried out in Section 3.4.

3.1 Motivation and Overview

The main idea of the method proposed by Kraft and Zölzer in [34] is to benefit from clues from both *spectral* and *temporal* representations of signals in order to accurately estimate multiple fundamental frequencies in polyphonic musical recordings. In short, a first set of $F0$ candidates is obtained from the spectral structure of the input signal, followed by a periodicity measure via temporal analysis, leading to the construction of a complementary set of candidates; lastly, both sets are combined so that eventual false candidates can be discarded, leading to a robust multiple $F0$ detection.

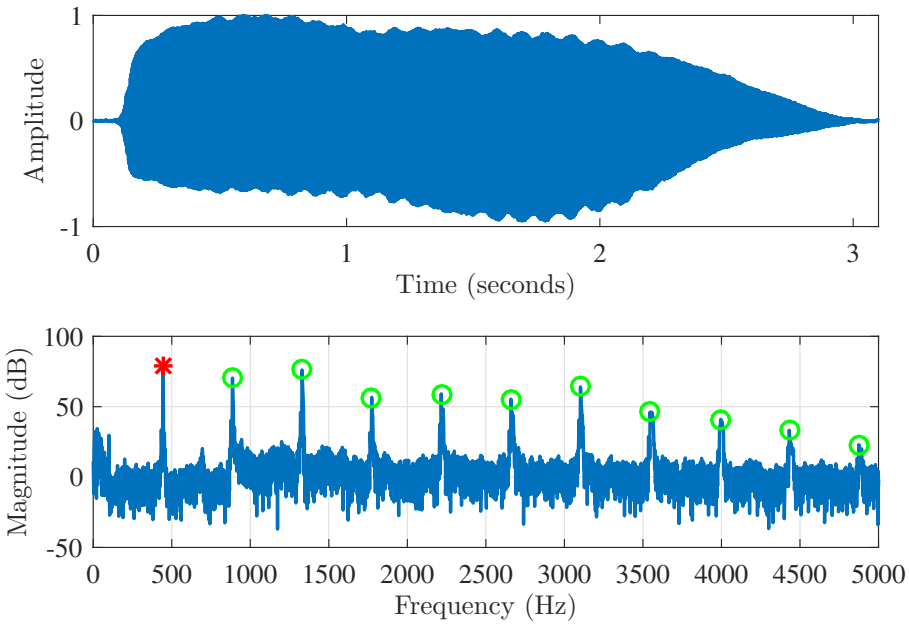

Figure 3.1: Waveform (top) and spectrum (bottom) of an A4 clarinet note. The fundamental frequency and its corresponding overtone partials are marked in the spectrum with an asterisk and circles, respectively.

Although characteristics such as spectral patterns (timbre) or acoustic waveforms may vary from one musical instrument to another, it can be stated that a sound of a single musical note consists mainly of one fundamental and various associated overtone partial components [59]. These main components, which are also called *harmonics*, are approximately regularly distributed in the spectral structure of the given musical note, and their amplitude values tend to decrease as they gain distance from the fundamental one; additionally, these components are characterised by a significant amount of energy around their respective bins, which correspond to peaky regions in the spectrum of the signal. A demonstration is given in Figure 3.1, where both waveform (top) and spectrum (bottom) of an A4 clarinet note are illustrated. As mentioned, the most prominent peaks represent the fundamental frequency (440 Hz) and its respective corresponding partials, which are respectively indicated on the graph by the asterisk and circles.

Furthermore, another way to characterise a fundamental frequency is by means of the periodicity of its waveform, since both concepts are inverse to each other (period is the inverse of frequency). This can be accomplished by means of a temporal analysis, and a strategy that works well is to analyse some kind of self-similarity

across time [60]. One simple way to detect such repetitive patterns in waveforms is to use the autocorrelation function (ACF), which in this context can be defined as [61]:

$$\mathrm{ACF}(m) = \frac{1}{W} \sum_{n=n_0}^{n_0+W-1-m} x[n]x[n+m], \tag{3.1}$$

where $x[n]$ is the input discrete waveform, n_0 indexes the sample at which the sum starts, W is the length of the window over which the ACF is computed, and the variable m denotes the time lag in the autocorrelation function.

For instance, when analysing the waveform of a single note, i.e. a monophonic sound with only one fundamental frequency to be estimated, the maximum of its autocorrelation function (apart from the value for zero lag) is expected to correspond to the fundamental frequency of the given sound. Analogously, when the input waveform comes from a polyphonic scenario, it is highly probable that the most prominent local maxima (again apart from the ACF value for zero lag) of its ACF correspond to the fundamental frequencies of the sound. Figure 3.2 shows the ACF of the same clarinet note exhibited in Figure 3.1. The highest peak, indicated by the asterisk, represents the fundamental period of the signal (approximately 2 ms). Since the signal is also periodic in integer multiples of the fundamental period, these multiples are also present in the autocorrelation function in the form of peaky regions, which are indicated by circles on the graph.

It is also important to make a brief consideration on the type of representation that will be adopted. As stated at the beginning of this section, Kraft's method is based on a temporal representation; it is indeed a time-domain approach, although not strictly so. There are alternative techniques which are based on splitting the signal through a filterbank before calculating the autocorrelation function. In this approach, an ACF is computed for each channel, and the resulting set of channel-wise ACFs is referred to as *multi-channel autocorrelation function* (MCACF). These individual functions may also be aggregated into a single representation, which is called *summary autocorrelation function* (SACF). The associated filterbank was originally proposed to simulate the behaviour of the human cochlea in [62], with subsequent

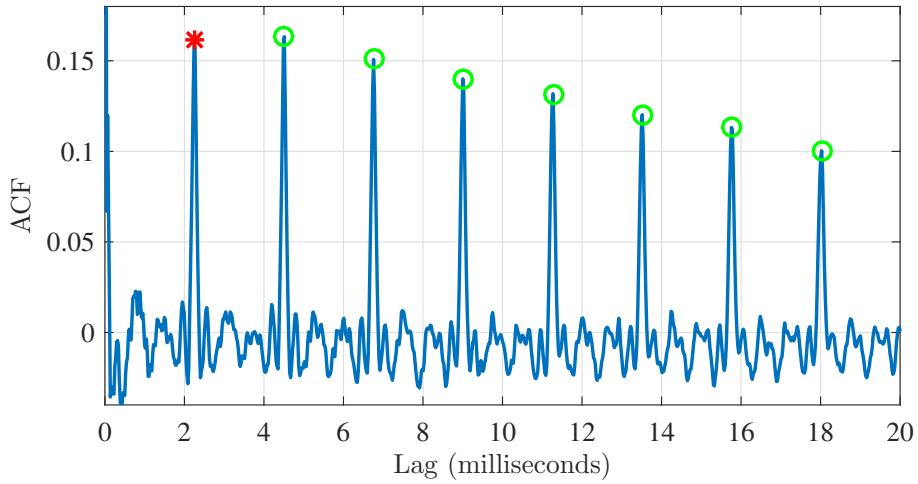

Figure 3.2: Autocorrelation function of an A4 clarinet note. The fundamental period and its corresponding subpartials are marked with an asterisk and circles, respectively.

improvements ([52, 63]) and simplifications ([33, 53]), in order to achieve lower computational complexity. Because of this spectral preprocessing, this variant of temporal methods is referred to as *spectrotemporal*, rather than just temporal. For a more detailed discussion on characteristics of spectral, temporal and spectrotemporal techniques for single/multipitch estimation with advantages/disadvantages of the three approaches, the reader may refer to [60]. Given this remark, and since Kraft's method computes an MCACF from the input signal, the representation that will be explored in the following sections is therefore defined more precisely as spectrotemporal.

Given this background on both spectral and spectrotemporal representations, it is possible to introduce the key idea of how Kraft's method works, as follows. On the one hand, for the spectral representation, a non-trivial decision on peaks to be selected should be made between the fundamental frequency f_0 and its associated partial components $2f_0$, $3f_0$ and so on, since these are the most prominent components of the spectrum. On the other hand, in the spectrotemporal representation, this decision is opposed: since a waveform that is periodic in T_0 is also periodic in $2T_0$, $3T_0$ and so on, this judgement should be made between the fundamental frequency f_0 and its *subharmonics*, which are $f_0/2$, $f_0/3$ and so forth. Therefore eventual faulty detections in peak detection for both representations are opposed.

To be then classified into a true estimated fundamental frequency, an element detected in one representation (*e.g.* the spectrotemporal one) must match a candidate in the other one (the spectral one, in this case).

In short, the main stages of the method are organised as follows. To begin with, the input signal is pre-processed by means of the short-time Fourier transform, from which a time-frequency representation, also called *spectrogram*, is obtained (see Subsection 3.2.1), in order to acquire a frame-wise representation. From this representation, both spectral and spectrotemporal analysis can be performed.

As regards the spectral peak selection (see Subsection 3.2.2), the first stage is to compute a *Tonalness* spectrum for each frame, representation introduced in [64] that indicates regions in the spectrum that are very likely to be tonal. Following this, a strategy for peak detection is then performed over this representation in order to separate tonal peaks from noisy ones.

As for the spectrotemporal analysis (see Subsection 3.2.3), firstly a pre-whitening algorithm is run over the input spectrogram, in order to equalise the spectral envelope for each frame, process which is also called spectral flattening. After that, the whitened spectra are split over a C-band fiterbank, followed by the computation of the multi-channel autocorrelation function from the individual ACFs of each filtered channel. Lastly, a peak selection procedure is applied to the MCACF in order to detect candidates for prospective fundamental frequencies.

The final step is the combination of potential candidates from both sets aforespecified in order to eliminate false positives. To do so, a decision criterion derived from the intersection operation[1] is imposed, but within a pitch range.

[1]In set theory, the intersection of two sets A and B is the set of elements that are in both A and B, but no other elements.

3.2 Method Description

3.2.1 Preliminary Processing

The pre-processing stage is derived from a sinusoidal analysis algorithm (see [65]). Audio signals obtained from music recordings can be represented as sequences of the musical notes which were played. Since these notes are not usually expected to be performed during the entire recording, analysing the spectrum of the whole signal is worthless, since it has no meaningful information in the context of this work. In other words, because musical notes have corresponding spectral frequency components, the spectral characteristics of a music signal vary in time, and such signals are called *non-stationary*. Therefore the input waveform, which is formerly in the time-domain, is converted into a time-frequency representation by means of the short-time Fourier transform (STFT) [66], so that spectral data of the signal can be revealed as it changes over time.

The steps of the STFT, which in this work is mathematically defined as

$$\mathrm{STFT}\{x[n]\} = \mathcal{X}(k,b) = \sum_{n=0}^{N_W-1} x[n+bN_H]\frac{w[n]}{N_W}e^{-2\pi jk/N_W}\,, \tag{3.2}$$

are listed as follows:

1. First the input signal $x[n]$ is multiplied by a shifting N_W-length normalised window function $w[n]$ (in this work, the Hann window function was adopted), in order to segment the entire waveform into small blocks, with contiguous blocks being overlapped by N_H samples, parameter which is also called *hop size*. The normalisation is performed in order to help choose parameters (which will be described in the following sections) that do not depend on the window length;

2. The N_{DFT}-point discrete Fourier transform (DFT) of each block is then computed, its outcome being denoted by $\mathcal{X}(k,b)$, representing the value of the k-th frequency bin of the b-th block;

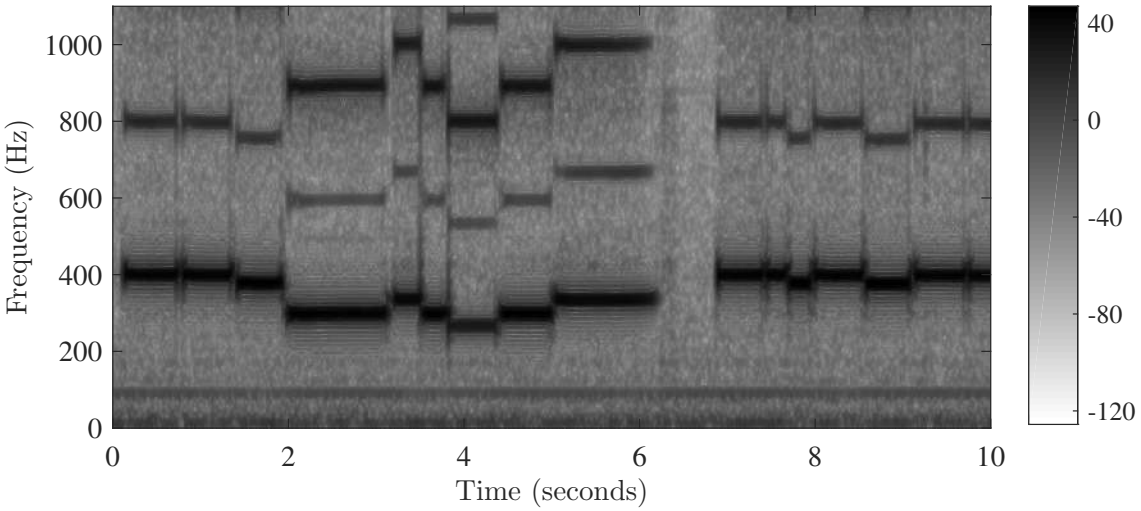

Figure 3.3: The spectrogram of a clarinet performance of Bach's composition Ach Gott und Herr.

Lastly, the absolute value $X(k, b) = |\mathcal{X}(k, b)|$, also called *magnitude*, is computed; this quantity expresses the intensity of the frequency components. A more concise discussion on the short-time Fourier transform, including formal definitions and demonstrations, analysis/synthesis models, and other considerations such as the role of the window function, can be found in reference texts on audio and music processing ([22, 66, 67]).

The magnitude $X(k, b)$ can be displayed by means of a two-dimensional representation, which is referred to as *spectrogram*. Figure 3.3 shows the spectrogram in dB of the clarinet part of the piece Ach Gott und Herr, one of the ten Bach chorales which constitute the Bach10 dataset (see Subsection 2.1.3). This demonstration, where the sampling rate of the input signal is $f_s = 44100$ Hz, was performed for window and hop sizes of 4096 and 441 samples respectively (corresponding to 93 and 10 ms), as well as for a 16384-point DFT.

3.2.2 Spectral Peak Selection

As previously discussed, for the spectral representation, a non-trivial decision must be made between the fundamental frequency and its respective partials. However, an initial step to estimate such potential tonal components from the spectrum is also necessary. In other words, genuine resonant components should be detected,

whereas the spurious ones, such as those which arise because of inevitable distortions caused by the time-frequency mapping or those caused by additive noise in the input signal, must be discarded. Therefore, the procedure of spectral peak selection can be divided into two steps: first, prospective tonal components are distinguished in the spectrum from noisy-induced ones; then, partials with irrelevant energy and potential false positives are removed from the set of selected peaks.

A wide range of approaches have been proposed in the literature over the years to address the task of spectral peak detection. Apart from threshold-based methods [68, 69], analysis-by-synthesis schemes [70] and model-based techniques [61] were also developed. A comparison of threshold-based schemes for peak detection can be found in [71].

1. Estimating tonal components: the Tonalness spectrum

The approach used in this work to detect prospective tonal candidates is to adopt the *Tonalness Spectrum*, introduced in [64]. This is a non-binary representation that is calculated over the signal spectrum, and it can be interpreted as the likelihood of a spectral bin to be a tonal or non-tonal component.

In this measure, a set of spectral features $\mathbb{V} = \{v_1(k,b), v_2(k,b), ..., v_{\mathrm{V}}(k,b)\}$ are computed from the signal spectrogram and combined to produce the overall tonalness spectrum:

$$\mathcal{T}(k,b) = \left(\prod_{i=1}^{V} t_i(k,b) \right)^{1/\eta}, \tag{3.3}$$

where $t_i(k,b) \in [0,1]$, which is referred to as the *specific tonal score*, is calculated over each extracted feature v_i according to:

$$t_i(k,b) = \exp\left\{ -\left[\epsilon_i \cdot v_i(k,b)\right]^2 \right\}, \tag{3.4}$$

and exponent η can be manually adjusted in the range $[1, ..., V]$, in order to adjust the computation in Eq. (3.3) between a simple product and a geometric mean, respectively; the association of Eq. (3.4) to Eq. (3.3) is similar to a simplified Radial

Basis Function network [72]. Factor ϵ_i is a normalisation constant that ensures that all features will equally contribute to the tonalness spectrum when combining them in Eq. (3.3), and it is obtained by setting in Eq. (3.4) the specific tonal score of the median over k of the feature in each block to 0.5. This produces the following formulation for the normalisation constant:

$$\epsilon_i = \frac{\sqrt{\log(2)}}{\overline{m_{v_i}}}, \tag{3.5}$$

where $\overline{m_{v_i}}$ is the average over all signal blocks (over all files, in the case that a dataset is being processed) of the median value $m_{v_i}(b)$ of feature i in the block b. The outcome of Eq. (3.4) can be interpreted as the probability that feature v_i presents a tonal component at bin k of block b.

The feature set comprises simple and established features, each one focusing on one particular aspect of a tonal component. These include a few that are purely based on information from the current magnitude spectrum of a block, such as frequency deviation, peakiness and amplitude threshold; some that are based on spectral changes over time, such as amplitude and frequency continuity; and another one that is based on the phase and amplitude of the signal, which is the time window centre of gravity. Moreover, the assessment of results in [64] revealed that, although the combination of features intuitively and empirically results in better scores than individual ones, combining more than three features did not achieve better representations. Lastly, it was also reported that combinations with a simple product, that is, adjusting η to 1 in Eq. (3.3), performed better than with the distorted geometric mean.

In the main reference of this chapter ([34]), the adopted tonalness spectrum is calculated from the combination of the *amplitude threshold* and *peakiness* features, and parameter η was set to 1. Therefore, in the present case, the expression in Eq. (3.3) can be simplified to:

$$\mathcal{T}(k,b) = t_{\mathrm{PK}}(k,b) \cdot t_{\mathrm{AT}}(k,b), \tag{3.6}$$

with $t_{\mathrm{PK}}(k,b)$ and $t_{\mathrm{AT}}(k,b)$ being the tonal scores of the peakiness and amplitude threshold, respectively.

The peakiness feature measures the inverse ratio between a spectral sample and its neighbouring bins, and it is defined as:

$$v_{\mathrm{PK}}(k,b) = \frac{X(k+p,b) + X(k-p,b)}{X(k,b)}, \tag{3.7}$$

where the distance p to the central sample should approximately correspond to the spectral main lobe width of the adopted window function when segmenting the signal, so that side lobe peaks can be avoided. In this work, the Hann window is adopted; therefore, p is set to approximately $2N_{\mathrm{DFT}}/N_{\mathrm{W}}$. What can be sorted out from Eq. (3.7) is that peakiness is a very local feature, and parameter p will determine to what extent this characteristic will be affected.

According to how Eq. (3.7) is defined, the direct form of the peakiness feature $v_{\mathrm{PK}}(k,b)$ is expected to be significantly close to zero for tonal peaks and high for non-tonal ones. In order to compare the feature with its prior original spectrum, it is better to illustrate the tonal score $t_{\mathrm{PK}}(k,b)$ rather than the feature in its pure form, since the latter is inverted and normalised to $[0,1]$ in Eq. (3.4). Figure 3.4 shows the peakiness tonal score along with the spectrum over which it was calculated, obtained from a frame which was extracted from the Bach piece Die Nacht (see Subsection 2.1.3) played by a clarinet and saxophone duet. As can be seen in the graph, the peakiness feature represents well how peaky the frequency bins are when compared with their surroundings, with even low magnitude peaks presenting a considerable tonal score.

The amplitude threshold feature measures the inverse ratio between the magnitude spectrum and an adaptive magnitude threshold, and it is defined as:

$$v_{\mathrm{AT}}(k,b) = \frac{r_{\mathrm{TH}}(k,b)}{X(k,b)}, \tag{3.8}$$

where $r_{\mathrm{TH}}(k,b)$ is a recursively smoothed version of the magnitude spectrum pro-

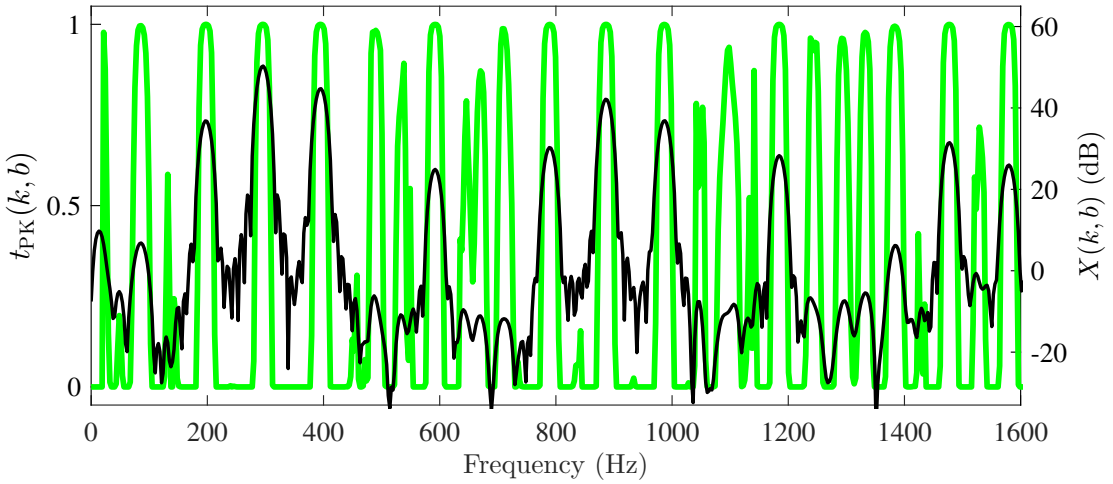

Figure 3.4: An example of the peakiness specific tonal score. The thicker and brighter line represents the feature of a frame from the piece Die Nacht, whose magnitude spectrum is indicated by the thinner and darker line.

duced by a single-pole low-pass filter:

$$r_{\mathrm{TH}}(k,b) = \beta \cdot X(k,b) + (1-\beta) \cdot r_{\mathrm{TH}}(k-1,b), \tag{3.9}$$

which is applied in both forward and backward directions in order to adjust the group delay, and $\beta \in [0,1]$ is a factor that is empirically tweaked.

The motivation of the amplitude threshold feature is that prominent peaks, which are potential candidates to be tonal components, are more likely to be above the threshold than less prominent ones, with this characteristic being measured by the ratio in Eq. (3.8). What can be sorted out from both Eqs. (3.8) and (3.9) is that the amplitude threshold is a more global feature when compared with peakiness, since the threshold $r_{\mathrm{TH}}(k,b)$ is calculated over the whole magnitude spectrum $X(k,b)$, *i.e.* a bin of $v_{\mathrm{AT}}(k,b)$ is affected by all bins from $X(k,b)$. Additionally, parameter β determines how similar to $X(k,b)$ will the threshold $r_{\mathrm{TH}}(k,b)$ be, with this similarity increasing as β gets close to 1. Lastly, it is also worth mentioning that parameter β, although empirically adjusted, should be inversely proportional to the spectrum size N_{DFT}, introduced in Subsection 3.2.1, since, as mentioned before, all bins in $X(k,b)$ are taken into account in a single bin of the threshold $r_{\mathrm{TH}}(k,b)$. Here, this parameter is empirically set to $\beta = 1500/N_{\mathrm{DFT}}$, which yields an approximate value

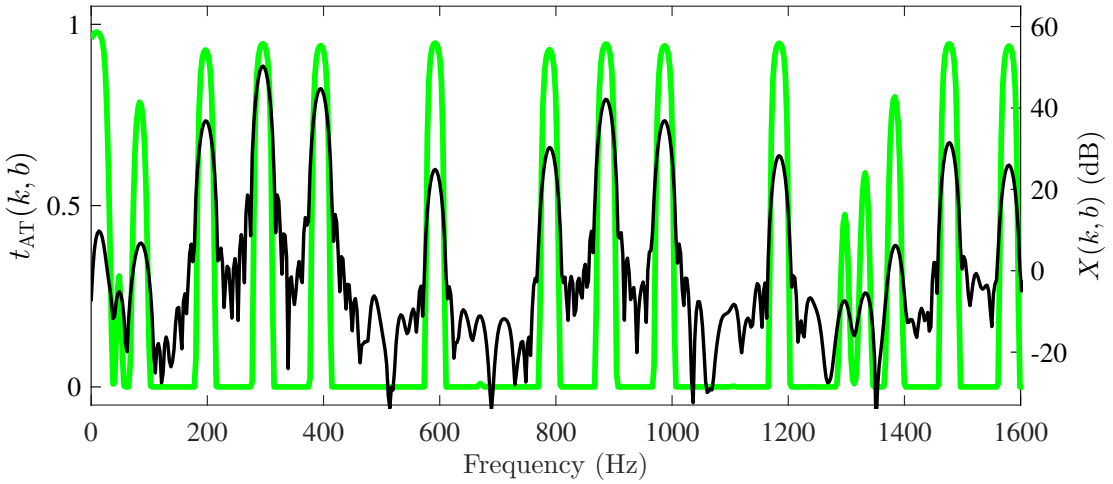

Figure 3.5: An example of the amplitude threshold specific tonal score. The thicker and brighter line represents the feature of a frame from the piece Die Nacht, whose magnitude spectrum is indicated by the thinner and darker line.

of 0.1.

Similarly to the peakiness demonstration in Figure 3.4, the tonal score of the amplitude threshold feature of that same frame is illustrated in Figure 3.5. For a 16384-point spectrum, the smoothing coefficient was adjusted to $\beta = 0.1$. As can be noticed from the chart, the amplitude threshold feature rejects more small peaks, even if some of these have a significant peakiness tonal score, when compared with Figure 3.4, validating the fact that the former is a more global feature where high-magnitude peaks affect smaller ones.

Therefore, the next step of the procedure is to, from the magnitude spectrogram $X(k, b)$ calculated in the preliminary processing stage described in Subsection 3.2.1, compute its tonalness spectrum $\mathcal{T}(k, b)$.

After computing the tonalness spectrum, all peaks k_i are selected in each block b, and those which fulfil the criterion

$$\mathcal{T}(k_i) \geq \mathcal{T}_{\text{TH}}, \tag{3.10}$$

are selected to a set of spectral peaks

$$\mathbb{P}_X = [k_1, ..., k_i, ..., k_K], \tag{3.11}$$

where $\mathcal{T}_{\text{TH}} \in [0, 1]$ is an empirically adjusted likelihood threshold, which aims to select only those peaks that are likely to be tonal, *i.e.* potential candidates to be true fundamental frequencies. Since the tonalness spectrum is a function of the peakiness and amplitude threshold features, the tuning of $\mathcal{T}_{\text{TH}}$ is highly dependent on the aforementioned parameters, which are the shift p of the neighbouring bins in Eq. (3.7) and the smoothing factor β in Eq. (3.9). In this work, a good balance between correct and rejected partials was achieved by setting $\mathcal{T}_{\text{TH}} = 0.6$. Although higher values for $\mathcal{T}_{\text{TH}}$ avoid wrong detections and hence improve the overall precision, they also compromise the recall measure by discarding true partials.

2. Removing irrelevant components and false positives

Since the tonalness spectrum evaluates peaky components and their surroundings, independently of their magnitudes, some insignificant and noisy peaks could present a high tonalness likelihood and thus be selected to $\mathbb{P}_X$. Hence, to discard these peaks from $\mathbb{P}_X$ the following local criterion is employed in each block:

$$X(k_i) \geq \gamma \cdot \max[X(k)], \tag{3.12}$$

and peaks that do not meet this criterion are removed. Parameter γ is a percentage factor that controls the rejection of peaks based on how small they are comparing with the global maximum, and it is set to 0.1% according to [34].

Lastly, the strength of each remaining peak $k_i \in \mathbb{P}_X$ is calculated by means of the salience function defined as

$$S_X(k_i) = \sum_{p=1}^{3} X(\hat{k}_p)^{0.25}, \tag{3.13}$$

which is a sum of the distorted magnitude values of the first three harmonic partials of each peak. The positions $\hat{k}_p$ are estimated around the partials in order to consider a certain quantity of inharmonicity. A constant search space Δ_k is applied in the surroundings of each overtone partial $k_p = p \cdot k_i$ and then the maximum $\hat{k}_p$ within

this range is considered in the salience function in Eq. (3.13). Additionally, the magnitude values are raised to the power of 0.25 so that low-energy components can have a significant contribution in the summation. At this stage, the frequency positions and magnitude amplitudes of the peaks in $\mathbb{P}_X$ and their corresponding partials $\hat{k}_p$ are refined via parabolic interpolation (see Appendix A). Implementation aspects also include imposing a limited frequency range, constrained by minimum and maximum values $F0_{\text{min}}$ and $F0_{\text{max}}$ (corresponding to k_{min} and k_{max}, respectively), since musical notes do not comprise the whole spectrum. Therefore all peaks whose corresponding frequencies lie outside this range are removed.

In the end, the conditions in Eqs. (3.10) and (3.12) can inevitably include some false positives in $\mathbb{P}_X$, so peaks whose salience values do not meet the criterion

$$S_X(k_i) > 0.1^{0.25} \cdot \max_{\forall k_i}[S_X(k_i)] \tag{3.14}$$

are finally discarded.

A demonstration of the selection of prominent tonal components using the combination of the tonalness spectrum with the conditions aforespecified is given in Figure 3.6, processed via Eq. (3.6) over the same peakiness and amplitude threshold tonal scores illustrated in Figures 3.4 and 3.5 respectively. By comparing the tonalness $\mathcal{T}(k)$ with its original magnitude spectrum $X(k)$, one can assert that the former is a powerful representation to estimate tonal components. The tonal peaks in the spectrum that appear above the tonalness threshold $\mathcal{T}_{\text{TH}}$, which was set to 0.7 and is represented in the graph by the horizontal dashed line, are selected according to Eq. (3.10). Following that, the peaks that do not meet the criteria in (3.12) and (3.14), which are indicated in the graph by circles, are discarded; the remaining that fulfil the conditions, highlighted with asterisks, are then selected to $\mathbb{P}_X$.

This concludes the spectral peak selection stage. The set of remaining peaks $\mathbb{P}_X$ and their respective salience values $S_X(k_i)$ of each block are stored and will be exploited in the peak matching stage in Subsection 3.2.4. In the following subsection, it is described how to estimate potential $F0$ candidates from the multi-channel

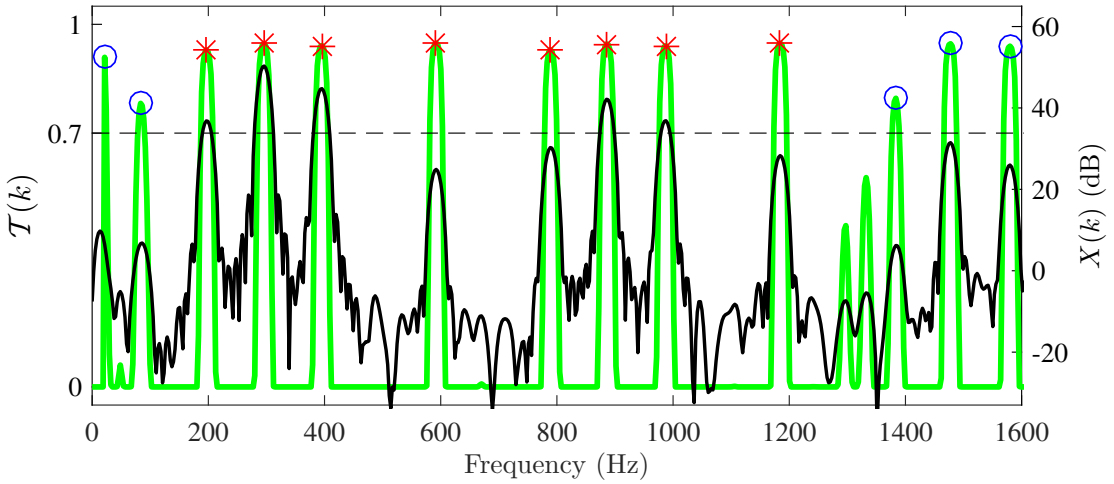

Figure 3.6: Example of peak selection using the tonalness spectrum. The thicker and brighter line represents the tonalness representation of a frame, whose magnitude spectrum is indicated on the thinner and darker line. The likelihood threshold is represented by the horizontal dashed line and the selected tonal peaks are marked with asterisks. The peaks above the likelihood threshold that do not meet the remaining conditions are marked with circles.

autocorrelation function.

3.2.3 MCACF Peak Selection

As opposed to the spectral peak picking strategy described in the previous subsection, for the spectrotemporal representation the decision must be made between the fundamental frequency and its respective subpartials. The strategy on MCACF peak selection can be divided into three stages, which are summarised as follows. To begin with, the spectrum of each frame is pre-whitened in order to remove short-time correlation from the signal [53]. Then, a filterbank is applied over each frame and the MCACF is calculated directly from this filtered version of the pre-whitened spectrogram. Lastly, a threshold and salience-based strategy is applied over the MCACF in order to select only the potential $F0$ candidates. At this point, the reader is invited to revisit the literature review in Section 2.2, for a brief summary of relevant works that exploit spectrotemporal representations to address the MPE problem.

1. Pre-whitening stage

Before computing the MCACF, the spectrum of each frame is flattened (or equalised) by means of a pre-whitening stage, where detected low-energy partials are amplified. From a performance point of view, this suppression of timbral information prior to the MCACF calculation makes the analysis more robust to different sound sources [3]. When viewed from an auditory perspective, this equalisation may be interpreted as the normalization of the hair cell activity level [73].

Spectral whitening can be achieved by many ways, as reported in the literature. In [53], this flattening is obtained via warped linear prediction (WLP) [74], which is an ordinary linear prediction applied over a frequency warped scale. In [3], the spectrum equalisation is obtained by estimating the rough spectral energy distribution followed by inverse filtering.

In this work, the spectrum of each frame is whitened by means of an algorithm that benefits from their respective spectral peaks $\mathbb{P}_X$ estimated in the last subsection. The steps of the algorithm are described as follows.

To begin with, a primary curve is interpolated through the spectral peaks $\mathbb{P}_X$, with this curve being referred to as the envelope $E'(k)$. For sake of simplification, in the rest of this work the block index b may be omitted. The interpolation[2] is carried out onto a logarithmic frequency axis, in order to favour the resolution in the low frequencies, which is commonly desired in the analysis of music signals.

Following that, the envelope $E'(k)$ is recursively smoothed similarly to Eq. (3.9), still on the logarithmic scale, in order to obtain the curve $E''(k)$:

$$E''(k) = \xi \cdot E'(k) + (1 - \xi) \cdot E''(k - 1), \tag{3.15}$$

with the smoothing parameter being adjusted to $\xi = 20/N_{\mathrm{DFT}}$ according to [34], and the single-pole low-pass filter being applied in both directions in order to compensate for group delay. The envelope $E''(k)$ is then interpolated back onto a linear frequency axis to produce the envelope $E(k)$.

[2]Here a shape-preserving piecewise cubic interpolation routine was used.

Lastly, the whitened spectrum is yielded by the division of the primary spectrum $X(k)$ by the last envelope:

$$X'_w(k) = \frac{X(k)}{E(k)} \tag{3.16}$$

and by normalising it so that both whitened and non-whitened spectra present equal power within the range limited by $F0_{\max}$:

$$X_w(k) = X'_w(k)\sqrt{\frac{\sum_{\kappa=0}^{k_{\max}} X(\kappa)^2}{\sum_{\kappa=0}^{k_{\max}} X'_w(\kappa)^2}}, \tag{3.17}$$

where $X_w(k)$ is the final whitened spectrum. Figure 3.7, adapted from [34], illustrates the pre-whitening stage, where all curves aforespecified are carefully depicted. The first graph reveals the original spectrum $X(k)$ with its respective detected peaks $\mathbb{P}_X$, along with the interpolated envelope $E'(k)$ and the recursively smoothed curve $E''(k)$. The second graph shows the original spectrum $X(k)$ compared with its whitened spectrum after normalisation $X_w(k)$. As expected, it can be seen that high-frequency components of $X(k)$ were amplified and the final spectrum $X_w(k)$ is well equalised.

2. MCACF computation

As mentioned in Section 3.1, before the calculation of the individual ACFs, the whitened spectrum $X_w(k)$ is split through a C-band filterbank, with each channel having a width of one octave starting from the smallest frequency $F0_{\min}$. First a group of filters with linear slopes is created

$$W'_c(k) = \begin{cases} \frac{4}{3k_c}k - \frac{1}{3} & \text{for} \quad \frac{1}{4}k_c < k < k_c \\ 1 & \text{for} \quad k_c \leq k \leq 2k_c \\ -\frac{1}{18k_c}k + \frac{10}{9} & \text{for} \quad 2k_c < k < 20k_c \\ 0 & \text{elsewhere,} \end{cases} \tag{3.18}$$

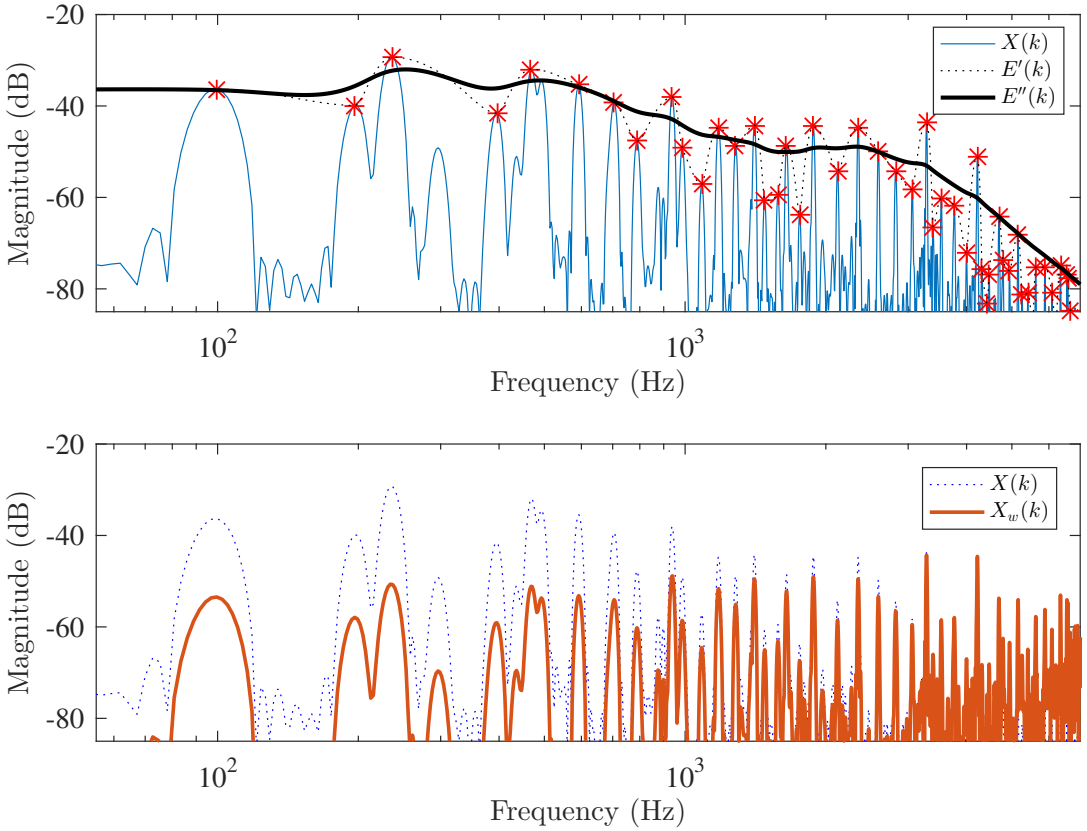

Figure 3.7: Illustration of the pre-whitening stage. In the top plot, the original spectrum $X(k)$ is indicated by the brighter and continuous line, and selected peaks are marked with asterisks; the interpolated envelope $E'(k)$ and its smoothed version $E''(k)$ are represented by the dotted and darker lines, respectively. Bottom plot compares the original spectrum $X(k)$ with its final normalised whitened version $X_w(k)$, indicated by the continuous and dotted lines, respectively.

where $c \in [0, C-1]$ is the current sub-band index and $k_c = 2^c k_{\min}$ is its lower boundary. Then the filters are normalised in order to compensate the higher energy in the superior octaves, caused by the increasing bandwidth as shown in Eq. (3.18):

$$W_c(k) = \frac{W'_c(k)}{\sum_{\kappa=0}^{N_{\mathrm{DFT}}-1} W'_c(k)}. \tag{3.19}$$

The slopes in Eq. (3.18) were empirically calculated in order to achieve an appropriated ACF for multiple $F0$ estimation. The adjustment of the slopes must take into account that, although being indispensable to remove high-frequency components in order to minimise the confusion between the sub-harmonics and the real fundamental frequencies, the presence of some partials also contribute to a more refined peak localisation in the ACF. Figure 3.8 recreates the original filterbank as

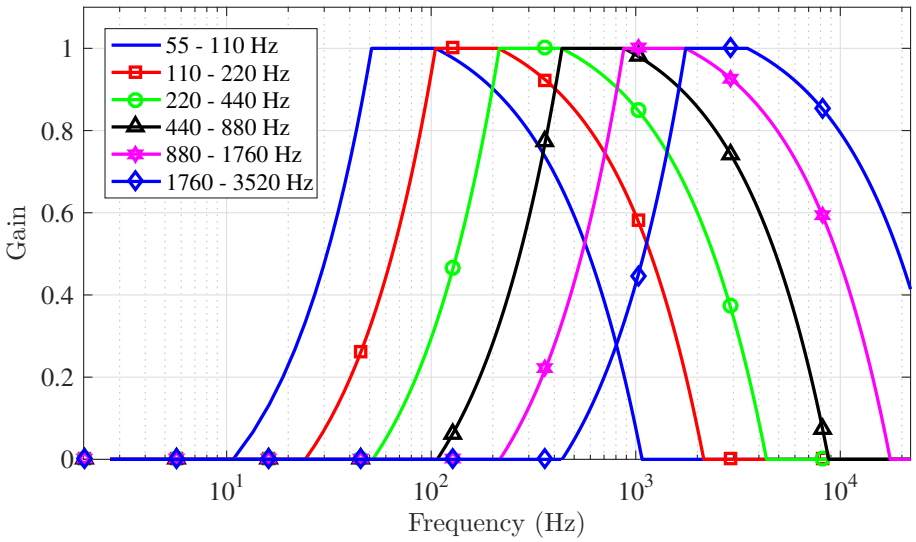

Figure 3.8: Magnitude responses of the proposed filterbank at which the whitened spectrum is split, calculated for six subbands starting from 55 Hz, with each band comprising one octave in the spectrum, as indicated in the legend.

proposed in [34], where the whitened spectrum $X_w(k)$ is split over five sub-bands starting from $F0_{\text{min}} = 55$ Hz, but with an additional sixth and last channel located after the fifth octave. The magnitude responses of the six bands are shown, each one comprising one octave in the spectrum.

To calculate the channel-wise ACFs, let us first consider the Wiener[3]-Khinchin[4]-Einstein[5] theorem:

$$\text{ACF}(m) = \text{IDFT}\Big\{\Big|\text{DFT}\{x[n]\}\Big|^2\Big\}, \tag{3.20}$$

which states that the autocorrelation function $\text{ACF}(m)$ of a signal $x[n]$ can be obtained by the inverse discrete Fourier transform (IDFT) of its respective power spectrum. By replacing the exponent 2 above by an adjustable parameter λ, it is possible to construct a *generalised autocorrelation function* [53]:

$$\text{ACF}(m) = \text{IDFT}\Big\{\Big|\text{DFT}\{x[n]\}\Big|^\lambda\Big\}, \tag{3.21}$$

which means that this metric can be calculated from a non-linearly distorted (or compressed) version of the original power spectrum. Related studies have shown

[3]Norbert Wiener, 1894 - 1964.
[4]Aleksandr Yakovlevich Khinchin, 1894 - 1959.
[5]Albert Einstein, 1879 - 1955.

that the standard autocorrelation function (*i.e.* by setting $\lambda = 2$) is actually sub-optimal for fundamental frequency estimation schemes [75], and choosing a proper value for λ improves the reliability and noise robustness for this task [53]. Examples of suggestions on the field of multiple $F0$ estimation include attempts with $\lambda = 0.67$ [53], 0.6 [33] and 0.5 [75]. It is also worth mentioning that this non-linear compression would not be achievable by using the time-domain formulation in Eq. (3.1), and gives the spectral computation of the generalised autocorrelation much more flexibility towards the design of a periodicity analysis mechanism. Additionally, the fast Fourier transform (FFT) and its inverse (IFFT) algorithms allow an efficient computation of the ACFs by means of Eq. (3.21) when compared with Eq. (3.1).

In this work, the exponent of Eq. (3.21) was adjusted to $\lambda = 0.5$. After denormalising the whitened spectrum $X_w(k)$ by the window length N_W inversely to Eq. (3.2) and applying the filters from Eq. (3.19), the final expression for the generalised autocorrelation in channel c can be obtained:

$$\mathrm{ACF}_c(m) = \mathrm{IDFT}\left\{ \left[X_w(k) N_W \right]^{0.5} W_c(k) \right\}, \tag{3.22}$$

and the set of resulting ACFs for each band constitutes the final multi-channel autocorrelation function.

3. Peak picking strategy

Similarly to the spectral peak selection, the MCACF representation contains lots of redundant and spurious information, and a proper strategy must be applied so that only strong $F0$ candidates are taken into account. To begin with, all peaks at time lag indices m_j^c are selected for all channels in each block, and those which fulfil the criterion

$$\mathrm{ACF}_c(m_j^c) > 0.001 \sum_c \mathrm{ACF}_c(0), \tag{3.23}$$

are selected to a set of MCACF peaks

$$\mathbb{P}_{\mathrm{ACF}_c} = [m_1^c, ..., m_j^c, ..., m_{M_c}^c], \tag{3.24}$$

where the adaptive condition in the right side of the expression in Eq. (3.23) is proportional to the sum of the channel-wise values of the zero-lag individual ACFs. Then the peaks m_j^c are constrained to their respective one-octave subbands, so elements of the set $\mathbb{P}_{\mathrm{ACF}_c}$ that do not meet

$$2^{-(c+1)} m_{\max} \geq m_j^c \geq 2^{-c} m_{\max} \tag{3.25}$$

are removed. At this point it is also important to remark that the maximal time lag $m_{\max}$ corresponds to the minimal frequency bin $k_{\min}$ (as well as $m_{\min}$ corresponds to $k_{\max}$), since their representations are inverse to each other.

Also, it is possible that a few bands do not carry enough information, mainly because of the flat slopes of the filters defined in Eq. (3.18), which can lead to redundancy between different bands. A solution is to discard from $\mathbb{P}_{\mathrm{ACF}_c}$ elements from bands c whose the maximum peak is considerably lower than the overall maximum value for all bands in the MCACF:

$$\max_{m_j^c \in \mathbb{P}_{\mathrm{ACF}_c}} [\mathrm{ACF}_c(m_j^c)] < 0.3 \max_{m < m_{\min}} [\mathrm{ACF}_c(m)]. \tag{3.26}$$

Lastly, the salience values of all elements $m_j^c \in \mathbb{P}_{\mathrm{ACF}_c}$ are calculated similarly to how spectral peaks were processed in Eq. (3.13). The MCACF salience function is defined as

$$S_{\mathrm{ACF}_c}(m_j^c) = \sum_{p=1}^{3} \mathrm{ACF}_c(\hat{m}_p), \tag{3.27}$$

where time lag $\hat{m}_p$ is the position corresponding to the maximum value within the range $\pm\Delta_m$ applied over the integer multiple $m_p = p \cdot m_j^c$ of its relative peak m_j^c. As opposite to Eq. (3.13), the input values of this salience function are not raised to any power; and in this analysis, negative values of the MCACF are not considered. In this stage, the time lag positions and amplitude values of the peaks in $\mathbb{P}_{\mathrm{ACF}_c}$ and their respective multiples $\hat{m}_p$ are refined via parabolic interpolation (see Appendix A), specially for short lags (which are associated with high fundamental

frequencies) where a semitone resolution is not accurate enough.

This concludes the MCACF peak selection stage. The set of peaks $\mathbb{P}_{\mathrm{ACF}_c}$ and their corresponding salience values $S_{\mathrm{ACF}_c}(m_j^c)$ of each frame are stored, and will be exploited together with the outcomes of Subsection 3.2.2 in the next subsection in order to finalise the multiple $F0$ detection method.

3.2.4 Peak Matching

In this subsection, both spectral (see Subsection 3.2.2) and spectrotemporal (see Subsection 3.2.3) representations are finally combined in order to estimate the multiple fundamental frequencies. As mentioned before, for the spectral peaks a decision must be made between the fundamental components and their respective multiples, whereas for the spectrotemporal peaks this decision is made regarding their corresponding sub-multiples. Therefore eventual detection errors in both domains are opposed to each other, and a strategy based on the intersection of sets $\mathbb{P}_X$ and $\mathbb{P}_{\mathrm{ACF}_c}$ is implemented.

The idea of fusing frequency and periodicity information was firstly proposed by Peeters in [76]. In this work, combinations of different functions derived from spectral and temporal representations are investigated in the context of single-pitch estimation. Emiya *et al.* [77] proposed later a parametric model that jointly benefits from a periodicity analysis and a spectral matching process for pitch detection of isolated piano notes.

In this method, the combination of temporal and spectrotemporal representations is achieved by multiplying their corresponding salience functions $S_X(k_i)$ and $S_{\mathrm{ACF}_c}(m_j^c)$ given by Eqs. (3.13) and (3.27), respectively. The corresponding peaks (k_i and m_j^c in the bin and time lag domain, respectively) of the aforementioned

salience functions are initially converted into MIDI notation:

$$Q_X(k_i) = 69 + 12\log_2\left(\frac{k_i f_\mathrm{s}/N_\mathrm{DFT}}{440\ \mathrm{Hz}}\right), \tag{3.28}$$

$$Q_{\mathrm{ACF}_c}(m_j^c) = 69 + 12\log_2\left(\frac{f_\mathrm{s}/m_j^c}{440\ \mathrm{Hz}}\right). \tag{3.29}$$

In the original work [34], the MIDI numbers $Q_X(k_i)$ and $Q_{\mathrm{ACF}_c}(m_j^c)$ are quantised to the nearest semitones (*i.e.* the values $\lceil Q_X(k_i) \rfloor$ and $\lceil Q_{\mathrm{ACF}_c}(m_j^c) \rfloor$ are calculated). Here, a strategy admitting non-integer MIDI values is implemented, therefore expanding the method to allow a more precise analysis of musical instruments with not-equal tempered notes or that can be played with frequency-modulation effects, such as vibrato and glissando.

The first step is to remove elements that are too close to each other. A threshold of one quarter tone (*i.e.* 0.5 in MIDI number) is imposed and, in case that two or more pitch candidates fall into this range, the one with higher salience value is preserved and the remaining candidates within this range are removed.

Therefore, a unique quarter tone mapping is created, where only the maximum salience value from the spectrum or MCACF in this range remains. Also, this step allows for the information of all channels of the MCACF to be summarised into a single set of values. The functions $S_{Q_X}(q)$ and $S_{Q_{\mathrm{ACF}}}(q)$ represent then the MIDI-to-salience mapping of the spectral and MCACF candidates respectively, and q indicates the MIDI variable.

As for the matching step, in [34] a simple product of the individual salience functions $S_{Q_X}(q)$ and $S_{Q_{\mathrm{ACF}}}(q)$ can then be calculated, since the candidates are truncated to the closest semitone value. Here, considering that any MIDI value is allowed, a match occurs when two candidates from each representation are close enough according to a threshold-based criterion. This threshold is manually tweaked and experimental results were obtained by setting this value to one quarter tone.

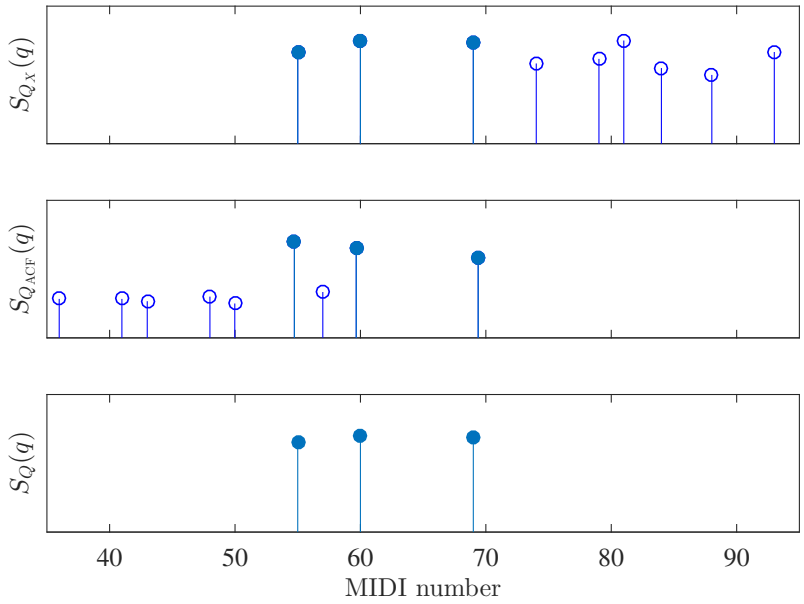

Figure 3.9: Illustration of the peak matching algorithm. The top plot shows the salience values of the spectral candidates; in the middle figure, the MCACF peaks are depicted; the bottom figure illustrate the final salience function, which is the product of both spectral and MCACF salience functions, respecting the threshold criterion. The true fundamental frequencies are marked with filled stems.

Therefore a final salience function can then be calculated:

$$S_Q(q) = S_{Q_X}(q) \cdot S_{Q_{\mathrm{ACF}}}(q), \tag{3.30}$$

reinforcing the fact that the product is evaluated only for candidates that are separated to each other by less than one quarter tone.

Figure 3.9, adapted from [34], shows a real example of the process of combining spectral and MCACF peaks. It can be seen that candidates of both representations contain a considerable quantity of false positives, represented by the harmonics and sub-harmonics of the true fundamental frequencies (indicated by the filled stems), which are removed after the matching strategy.

Furthermore, a final threshold must be imposed so that detections with significantly low or zero salience values are lastly removed:

$$S_Q(q) > S_{\mathrm{TH}}, \tag{3.31}$$

where the salience threshold S_{TH} is obtained by empirically adjusting it so that the maximum F-measure (defined in Subsection 2.1.4) is achieved.

Lastly, it is important to mention that the goal of this strategy of picking candidates from both representations is to discard false positives; there is also the problem of missing detections (*i.e.* the false negatives). Therefore, in order to minimise the occurrence of missed true pitch candidates, it is convenient to set relaxed thresholds in Eqs. (3.10), (3.12), (3.14), (3.23) and (3.26), so that all true fundamental frequencies appear as potential candidates in both representations.

This concludes the peak combination stage and the description of the multi-pitch estimation method. In the next subsection, the algorithm is evaluated on annotated music signals and the results is discussed.

3.3 Results

In this section, the method described in this chapter for multiple fundamental frequency estimation is evaluated on the datasets introduced in Subsection 2.1.3. First, the influence of the polyphony level is analysed, then the datasets are evaluated generally. For both analyses, comparison with literature methods are carried out.

3.3.1 Influence of the polyphony level

Since any audio signal with more than one concurrent pitched sound is considered polyphonic, it is important to analyse how the implemented method for multiple fundamental frequency estimation performs according to the degree of complexity of an input audio signal. One way to measure complexity of audio signals is to use the prior information of the maximum number of concurrent sounds. Therefore a signal can be considered more complex as the polyphony number increases.

To investigate the dependency of different polyphony levels, the Bach10 and MIREX datasets are evaluated. The TRIOS collection is not suitable for this analysis since all tracks comprise polyphonic piano parts, and it is more intuitive to

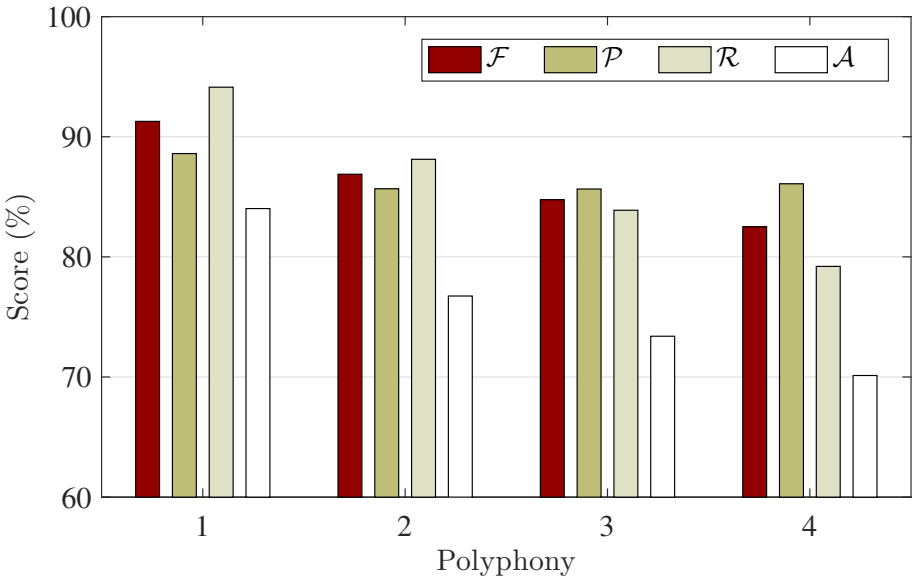

Figure 3.10: Multiple fundamental frequency estimation results for the Bach10 dataset (F-measure, precision, recall and accuracy) on a polyphony level basis.

evaluate this dataset using the full polyphony tracks.

Figure 3.10 shows bar plots of the F-measure ($\mathcal{F}$), precision ($\mathcal{P}$), recall ($\mathcal{R}$) and accuracy ($\mathcal{A}$) results for the Bach10 collection for each polyphony number, as indicated on the horizontal axis; the vertical axis represent the overall score in percentage values. A very noticeable trend that can be extracted from the plots is that F-measure, recall and accuracy scores decrease as the maximum number of concurrent sounds grows. Moreover, it can be seen that the precision score remains approximately constant at around 86% for all polyphonic combinations (levels 2, 3 and 4), being slightly smaller than precision obtained from the monophonic set, which figures at approximately 88.6%.

Table 3.1 compares the results obtained from both modified (see Subsection 3.2.4) and original ([34], also implemented by the author) methods. The term NQ in the table stands for *non-quantised*. The table also presents the scores obtained from the method proposed by Duan *et al.* in [17] (a ready-to-run implementation is publicly available[6]). The highest scores per metric (F-measure, precision, recall and accuracy) are marked bold[7]. As it can be seen in the Table 3.1, the scores obtained

[6] `http://www2.ece.rochester.edu/projects/air/resource.html`

[7] This will be a common practice within this work, as will be revealed in the next tables.

Method	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	**91.3**	88.6	**94.1**	**84.0**
Kraft [34]	91.0	**89.1**	93.1	83.7
Duan [17]	76.6	68.1	87.6	61.2

(a) Bach10 dataset results for polyphony 1.

Method	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	**86.9**	**85.7**	**88.1**	**76.7**
Kraft [34]	86.3	**85.7**	86.9	75.9
Duan [17]	73.5	64.2	85.7	56.9

(b) Bach10 dataset results for polyphony 2.

Method	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	**84.8**	**85.7**	**83.9**	**73.4**
Kraft [34]	84.1	**85.7**	82.6	72.5
Duan [17]	71.8	63.7	82.2	55.5

(c) Bach10 dataset results for polyphony 3.

Method	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	**82.5**	**86.1**	**79.2**	**70.1**
Kraft [34]	81.9	86.0	78.2	69.3
Duan [17]	70.9	65.1	77.9	54.9

(d) Bach10 dataset results for polyphony 4.

Table 3.1: Detection evaluation results and comparison grouped according with polyphony number for the Bach10 dataset.

from the modified method NQ-Kraft are slightly better than those yielded by the original one for all polyphony degrees, with the overall F-measure scores exceeding the original method by approximately 0.6% for all combinations with polyphony level greater than or equal to two. It is also worth mentioning that the personal implementation of Kraft's method achieved scores almost identical to those reported in the original paper. Moreover, the NQ-Kraft method performed significantly better than that of Duan *et al.*, surpassing all scores, except for recall, by at least 11% for all degrees of polyphony.

The bar graph in Figure 3.11 presents the evaluation results for the MIREX collection for each polyphony number. Since this dataset is obtained from a quintet recording, the degrees of polyphony now vary from one to five. As in Figure 3.10, the F-measure, recall and accuracy results decrease as the polyphony level increases. It can also be noticed that the precision score once more figures roughly constant for all degrees of polyphony.

Table 3.2 compares the results obtained from both modified and original methods, as well as with the algorithm proposed by Duan *et al.* [17]. As in Table 3.1,

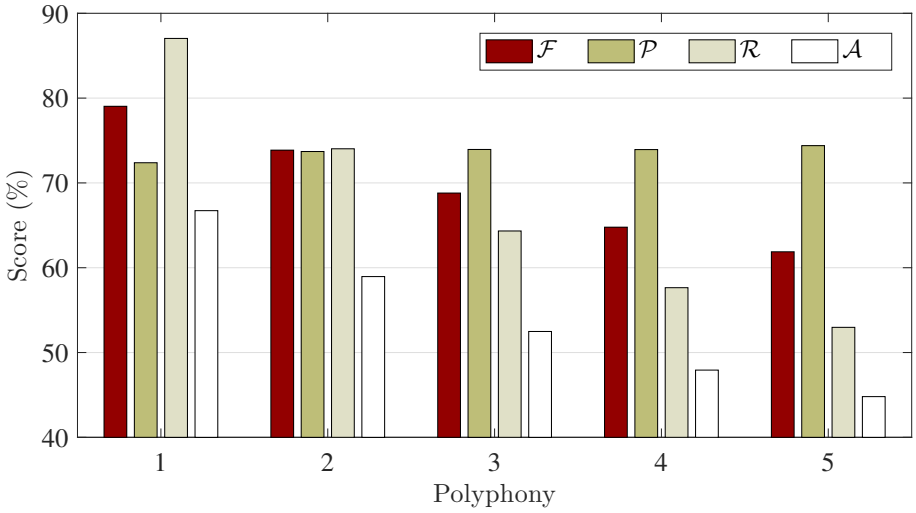

Figure 3.11: Multiple fundamental frequency estimation results for the MIREX dataset (F-measure, precision, recall and accuracy) on a polyphony level basis.

the modified method NQ-Kraft performed slightly better than the original one with all polyphony degrees for the MIREX dataset. Interestingly, it can be also seen that, as the polyphony level increases, the difference between performances of the modified and original algorithm is more noticeable for the F-measure score, figuring at approximately 1.5% for the full combination of instruments. It is important to remark that, for the MIREX dataset, the personal author's implementation of Kraft's method obtained scores a few percent worse than those reported in the original reference, and this consequently has affected the evaluation of the NQ-Kraft algorithm. Furthermore, the NQ-Kraft method yielded considerably better overall F-measure scores than those obtained by the technique proposed by Duan *et al.*, with the difference between performances presenting a rough downward trend as the number of concurrent sounds increases.

In order to conclude this subsection and summarise the evaluation, a few more comments and discussion are carried out as follows. One can notice that, as stated on the first impressions of results from Tables 3.1 and 3.2, the implemented algorithm NQ-Kraft shows a slight advantage over the original method proposed by Kraft and Zölzer [34]. This rough improvement on the results was already expected, since the non-quantised scheme benefits the estimation only in the case when two corresponding spectral and MCACF peaks would fall into different semitone ranges

Method	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	**81.0**	72.4	**89.0**	**68.7**
Kraft [34]	80.3	**72.6**	88.6	66.6
Duan [17]	67.9	63.2	73.3	53.3

(a) MIREX dataset results for polyphony 1.

Method	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	**75.9**	**73.7**	**76.0**	**61.0**
Kraft [34]	75.0	73.6	75.7	59.4
Duan [17]	63.5	65.1	61.9	46.4

(b) MIREX dataset results for polyphony 2.

Method	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	**71.8**	**73.9**	**67.3**	**55.5**
Kraft [34]	70.5	73.6	66.9	53.9
Duan [17]	60.2	65.4	55.7	42.8

(c) MIREX dataset results for polyphony 3.

Method	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	**70.0**	72.5	**65.3**	**53.4**
Kraft [34]	68.6	**73.7**	64.5	50.3
Duan [17]	57.2	65.5	50.7	40.0

(d) MIREX dataset results for polyphony 4.

Method	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	**64.2**	**74.4**	**57.1**	**49.8**
Kraft [34]	62.7	73.5	55.6	47.0
Duan [17]	54.8	65.8	46.9	37.7

(e) MIREX dataset results for polyphony 5.

Table 3.2: Detection evaluation results and comparison grouped according with polyphony number for the MIREX dataset.

when using the original method. For instance, two supposed-to-be-matched spectral and MCACF peaks with respective MIDI numbers 69.4 and 69.6 would fall into the semitones 69 and 70 after rounded to the nearest integer, even though they should have been matched together in order to yield a correct pitch estimation in a specific frame or sequence thereof.

Another expected result that can be clearly seen is that the overall performance of all MPE algorithms, including the implemented, original and publicly available ones, decreases as the sound mixtures get more complex. The main challenge in MPE is to develop robust algorithms that can produce satisfactory results even when the input sound has a high degree of polyphony. Lastly, it can be sorted out from the obtained results (and this is easily noticeable by looking at the bar graphs) that, for both datasets, the precision score remains roughly constant independently of the polyphony level. This means that the strategy for false positive rejection is quite robust and therefore lightly affected by the number of concurrent sounds.

3.3.2 Complete datasets

As reported in literature, the great majority of reference works assess only the full polyphony datasets. Therefore, in order to make possible a meaningful comparison of the implemented methods with state-of-art ones, this subsection focuses on the evaluation of the complete target datasets. Here the TRIOS collection (the reader may refer to Subsection 2.1.3), which is a more complex compilation of five sound mixtures, is first evaluated.

As seen in Section 2.2, a vast number of MPE algorithms have been proposed in the literature. However, some of them are not based on a "blind" approach, that is, some of them use prior information about the input signal such as specific musical instrument models [39] or the exact number of instruments [40]; some also use post processing refinement steps like note [41] or timbre [5] tracking. In other to make a fair comparison of the NQ-Kraft algorithm (which is blind and uses no refinement steps) with state-of-art techniques, only those that do not use neither

Algorithm	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft*	82.5	86.1	79.2	70.1
Kraft and Zölzer ([34]*)	81.9	86.0	78.2	69.3
Cheng *et al.* [30]†	80.7	81.9	79.6	67.7
Benetos and Dixon [57]‡	78.7	78.4	79.1	65.0
Kraft and Zölzer [33]†	74.0	69.3	79.3	58.7
Duan *et al.* [17]§	70.9	65.1	77.9	54.9
Benetos *et al.* [43]‖	68.4	61.6	76.8	51.9
Sigtia *et al.* [31]†	65.2	62.8	67.8	48.4
Benetos and Weyde [32]†	65.0	57.3	75.1	48.2
Klapuri [50]¶	61.9	60.0	64.0	44.9
Tolonen and Karjalainen [53]¶	61.4	61.5	61.2	44.2

Table 3.3: Detection evaluation results and comparison for the Bach10 dataset.

previous information about the music nor post processing stages are taken into account in this subsection.

Table 3.3 shows the multi-pitch detection results for the complete Bach10 collection. Besides the two methods compared in Tables 3.1 and 3.2, eight other reference algorithms are also investigated, each one carefully reviewed in Section 2.2. The techniques are sorted in descending order according to their respective F-measure scores. The superscript legends on the table have the following meanings: asterisk * stands for the algorithms implemented by the author; the method labelled with section sign § was evaluated by the author via publicly available code; dagger † indicates the methods whose results were reported directly in the original paper; the method marked with the double dagger ‡ was evaluated by the authors of [30]; the algorithm indicated with double bar ‖ was run by the authors of [33] via the original code; and the methods indicated by the pilcrow ¶ were implemented and run by the authors of [33].

Table 3.3 shows that the NQ-Kraft algorithm, along with the original one [34], yield generally better scores than the other nine approaches, attaining approximately 82.5% for the F-measure score. Apart from those, the method that best performed is the one proposed by Cheng *et al.*, achieving an F-measure score approximately 1.8% lesser than NQ-Kraft, followed by Benetos and Dixon's algorithm, which yields

Algorithm	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
Cheng *et al.* [30]†	64.9	76.9	56.1	48.0
NQ-Kraft*	64.2	74.4	57.1	49.8
Benetos *et al.* [43]‖	63.9	62.0	65.9	46.9
Kraft and Zölzer [34]*	62.7	73.5	55.6	47.0
Benetos and Dixon [57]‡	62.5	74.8	53.7	45.5
Vincent *et al.* [36]†	62.5	-	-	-
Kraft and Zölzer [33]†	61.6	58.3	65.3	44.5
Benetos and Dixon [38]†	58.4	55.0	62.2	47.8
Duan *et al.* [17]§	54.8	65.8	46.9	37.7
Klapuri [50]¶	51.0	50.5	51.5	34.2
Tolonen and Karjalainen [53]¶	41.4	40.5	42.3	26.1

Table 3.4: Detection evaluation results and comparison for the MIREX dataset.

a score of 78.7 % for F-measure as well. It is interesting to note that, among all methods, NQ-Kraft is the one with the highest precision score, that is, it has the best performance in terms of false positive rejection.

Table 3.4 shows the multiple fundamental frequency estimation results for the complete MIREX dataset for eleven different algorithms, including those shown in Table 3.2. The methods evaluated in this table that are not displayed in Table 3.3 are also revised in Section 2.2. Like in the previous table, methods are sorted in descending order according to their respective F-measure scores and the superscript legends follow the same labels.

The system that best performed among all was the one proposed by Cheng *et al.*, achieving an F-measure score of 64.9% as reported in the original work, followed by NQ-Kraft algorithm, which obtained a score of approximately 64.2%. It is important to mention again that the author's implementation of the original Kraft's method yielded lower scores than those reported in [34]. Nevertheless, the results obtained using the NQ-Kraft system are still considered satisfactory, when compared with other reference methods. The technique proposed by Benetos *at al.* figures as the third best method, achieving an F-measure score of 63.9%. The precision score achieved via NQ-Kraft also figures as one of the best reported, hence it can be inferred that the proposed system still has a satisfactory rate of false

Algorithm	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
Benetos *et al.* [43]$^{\|}$	57.7	68.6	49.8	40.6
NQ-Kraft*	56.5	78.3	43.3	38.5
Kraft and Zölzer [34]*	55.8	82.0	42.3	38.6
Kraft and Zölzer [33]†	54.5	58.8	50.8	37.5
Duan *et al.* [17]§	45.8	59.8	37.1	28.1
Klapuri [50]¶	45.7	52.3	40.5	29.6
Tolonen and Karjalainen [53]¶	43.0	48.0	38.8	27.3

Table 3.5: Detection evaluation results and comparison for the TRIOS dataset.

positive rejection for the MIREX collections, which has a higher degree of polyphony than that of the Bach10 dataset.

The TRIOS dataset was released more recently and fewer works have been evaluated over it in the context of AMT or MPE, as can be seen in Table 3.5. This collection is also more complex than Bach10 and MIREX datasets, since all of their music signals contains a piano track, a musical instrument with a high degree of polyphony. In fact, experimental results showed that signals from this dataset can reach up to 13 concurrent pitches in a same time frame.

As in the previous tables, methods are sorted according to their respective F-measure scores, and the superscript legends have the same meaning. From Table 3.5 it is possible to see that NQ-Kraft algorithm achieved the second best $\mathcal{F}$ score, figuring at approximately 56.5%, just around 1.2% below the score yielded via the method proposed by Benetos *et al.* [43]. The original Kraft's algorithm [34] implemented by the author appears as the third best method is in evaluation, with an F-measure score of about 55.8%, followed by previous work of Kraft and Zölzer [33], which achieved 54.5% for the same metric. The remaining techniques performed considerably worse, with $\mathcal{F}$ scores figuring at least 10% lower than that obtained by running the NQ-Kraft algorithm.

By comparing the modified NQ-Kraft method with the original work [34], the overall F-measure score of the former was slightly better than the latter, as expected. Interestingly, NQ-Kraft performed worse in terms of precision, with this score figur-

ing approximately 3.7% lesser than that obtained by running the original method. This indicates that, for the TRIOS dataset, the non-quantising approach does not reject as many false positives as the quantising one does.

In order to summarise the evaluation of the complete datasets, a few more observations are drawn. For all evaluations, as seen in Tables 3.3, 3.4 and 3.5, the NQ-Kraft algorithm yielded satisfactory results, figuring among the best methods that neither benefits from prior information of the inputs nor uses any post processing stages, specially for the Bach10 collection, over which it was ranked as the best one.

Also, another expected result was the downward trend in the overall performance according to the complexity of the dataset, that is, the average number of concurrent sounds. Among the collections, NQ-Kraft algorithm achieved its best results for the signals in Bach10 set, which is the less complex dataset with its four monophonic instrument recordings. Following that, the evaluation of MIREX collection resulted in the second best scores, with this dataset being constructed using five different monophonic woodwind instruments. Lastly, the worse scores were achieved by evaluating the TRIOS dataset, whose tracks contain piano recordings and hence are classified as the most complex sound mixtures.

3.4 Conclusion and final considerations

In this chapter, the main method for multiple fundamental frequency estimation approached in this work was explained in details, and it was proposed its modification to allow non-integer MIDI values, thus enabling the analysis of both non-tempered and frequency modulated notes. Benchmark datasets were employed in order to assess the performance of the method.

Overall, the modified method NQ-Kraft performed better than its original version, and the results were considered satisfactory, figuring among the best ones when comparing with reference methods from the literature. One drawback of the presented method is the considerable quantity of parameters that can be only adjusted

experimentally. A significant number of empirical tests were performed by the author within implementation stages, which proved to be very time consuming, and these experiments showed a strong dependency on the parameter values, which have a considerable degree of freedom.

In the next chapter, refinement algorithms are proposed to be integrated into the NQ-Kraft method, in order to improve the quality of the estimation of concurrent fundamental frequencies.

Chapter 4

User interaction and post-processing refinements

Results shown in last chapter reveal that even state-of-art algorithms still perform far from a perfect multi-pitch estimation. In fact, the performance of the evaluated systems is not satisfactory for applications that require a high degree of accuracy. This chapter aims to benefit from user interaction and post-processing steps so that the NQ-Kraft method can yield better scores.

A recent study carried out by Benetos *et al.* [6] on challenges and future directions of automatic music transcription claims that current systems have apparently reached a performance limit, and one of the proposed solutions of this problem is to adapt existent methods for user-assisted (or semi-automatic) approaches. In Section 4.1, a refinement of NQ-Kraft algorithm is proposed in order to allow for the system to benefit from prior information regarding the maximum number of concurrent sounds.

Following that, two post-processing algorithms are implemented in Sections 4.2 and 4.3. The first one benefits from neighbouring frames to refine fundamental frequency estimates in the actual frame whereas the second one performs a note tracking algorithm to both remove detection errors and correct note discontinuities. Lastly, final considerations are drawn in Section 4.4 in order to conclude the chapter.

4.1 Polyphony informed

The salience function, introduced in Eq. (3.13), is a measure of how strong an $F0$ candidate is, since it is defined as a weighted sum calculated over the overtone partials of a specific peak. Taking this into account, a simple strategy using salience functions is proposed to take advantage of prior information about the maximum number of concurrent sounds in the input music signal. It is worth mentioning that this strategy does not add any substantial computational complexity to the method described in Chapter 3, since it benefits from the salience functions that are already computed within the main algorithm.

Let $S(x)$ be a generic salience function calculated for each candidate over an also generic representation indicated by the variable x. Given that P is the pre-informed maximum polyphony number of the input signal, the salience values corresponding to the peak candidates of each frame are initially sorted in descending order. Then only candidates whose salience functions lie among the highest P values are kept, hence discarding the remaining ones.

It turns out that the system described in Chapter 3 explores three different salience functions, which are defined in Equations (3.13), (3.27) and (3.30). In order to investigate which salience function best suits this algorithm, an experiment is run for the Bach10 dataset varying the polyphony number of the signals and evaluating the scores for three scenarios, each one adopting a different salience value. Then the salience function that produces the best scores is chosen for the algorithm.

Table 4.1 shows the results for the experiment described in Chapter 3. The first row of each sub-table shows the scores yielded via the NQ-Kraft method with no prior information about the maximum polyphony number (same scores as shown in Table 3.1), whereas the three bottom rows reveal the scores obtained by exploiting each respective salience function in the polyphony informed approach, as indicated on the first column of each sub-table.

It can be seen that the final salience S_Q achieved the best scores when comparing to the clean NQ-Kraft method, with the second best results being obtained via the

Method	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	91.3	88.6	94.1	84.0
S_{Q_X}	**94.3**	**94.7**	**93.9**	**89.4**
$S_{Q_{\text{ACF}}}$	93.9	94.3	93.4	88.6
S_Q	**94.3**	**94.7**	**93.9**	89.3

(a) Bach10 dataset results for polyphony 1.

Method	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	86.9	85.7	88.1	76.7
S_{Q_X}	87.3	90.4	84.4	77.7
$S_{Q_{\text{ACF}}}$	88.8	92.0	85.9	80.0
S_Q	**89.0**	**92.2**	**86.1**	**80.4**

(b) Bach10 dataset results for polyphony 2.

Method	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	84.8	85.7	83.9	73.4
S_{Q_X}	84.8	86.2	83.5	73.6
$S_{Q_{\text{ACF}}}$	84.9	**86.3**	**83.6**	73.7
S_Q	**85.0**	**86.3**	**83.6**	**73.8**

(c) Bach10 dataset results for polyphony 3.

Method	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	82.5	86.1	**79.2**	70.1
S_{Q_X}	**82.6**	**86.6**	79.0	**70.3**
$S_{Q_{\text{ACF}}}$	82.5	86.5	78.9	70.2
S_Q	**82.6**	**86.6**	78.9	**70.3**

(d) Bach10 dataset results for polyphony 4.

Table 4.1: Influence of different salience functions for the informed polyphony scheme in the evaluation of the Bach10 dataset.

spectral salience S_{Q_X}, followed by the MCACF salience $S_{Q_{\text{ACF}}}$, which achieved the worst values. As a result, the final salience function S_Q will be integrated into this algorithm. An important pattern seen on the table is that the proposed refinement performs better for signals with a small degree of polyphony. The overall performance improvement, taking into account the experiment using the final salience function S_Q, for the F-measure score is approximately 3% and 2.1% for polyphony levels 1 and 2 respectively, whereas for maximum number of concurrent pitches 3 and 4 the improvement on this score is around 0.2% and 0.1%, respectively. It is also important to mention that, as expected, the precision scores improves. This happens because the approach of this refinement is discarding only potential faulty detections, which affects directly in the computation of the precision score (see Eq. (2.2)).

The improvement for the complete dataset, *i.e.* for full polyphony, is still very slight. Although the proposed strategy indeed removes incorrect estimations, there is a drawback associated with its simplicity. One of the main difficulties in MPE

methods is to eliminate candidates that are actually the second or third harmonic of a strong $F0$ candidate, which in the context of this work is translated as a candidate with a corresponding high salience value. As a result, the polyphony informed refinement receives the set of pitch estimates with potential errors caused by the harmonics of true pitches, and these harmonics can also present a significant salience value depending on the strength of their respective fundamental frequencies. In [3], Klapuri proposes both an iterative and a joint strategy to work with salience functions in MPE, and future works regarding this section aim to integrate Klapuri's approach in order to produce a more efficient algorithm for polyphony informed MPE.

4.2 Neighbouring frames refinement

As seen in last section and in Chapter 3, multi-pitch estimation can produce several types of errors, either produced by incorrect pitch estimates or those resulted from the non-detection of true pitches. This first post-processing refinement was proposed by Duan *et. al.* in [17], and its key idea is to use $F0$ candidates from neighbouring frames to refine $F0$ estimates in the current frame. It is assumed that pitches of musical signals are locally stable in the order of approximately 100 ms, and departing from this an algorithm is proposed to remove potential detection errors and reconstruct estimates which have not been detected.

The first step is to build a weighted histogram $W(b, q)$ in the frequency domain for each frame b. There are 63 bins in $W(b, q)$, indicated by the variable q, corresponding to the 63 MIDI numbers from 33 to 95 (which correspond to the semitones from notes A1 to B6, respectively). In order to construct the histogram $W(b, q)$, a triangular weighting function $w_t(b)$ in the time domain is applied onto a neighbourhood of b, limited by a radius of R frames, in two different ways.

The function $w_t(b)$ is firstly applied to the vector $P(b)$, defined as the number of estimated pitches per frame. The result of this computation is rounded to the nearest integer, yielding a refined polyphony estimate per frame $\hat{P}(b)$. Then, the

	Metric (%)			
Method	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	**91.3**	**88.6**	94.1	**84.0**
NBF	90.9	86.4	**95.9**	83.4

(a) Bach10 dataset results for polyphony 1.

	Metric (%)			
Method	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	86.9	**85.7**	88.1	76.7
NBF	**87.0**	83.7	**90.7**	**77.0**

(b) Bach10 dataset results for polyphony 2.

	Metric (%)			
Method	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	84.8	**85.7**	83.9	73.4
NBF	**85.5**	83.8	**87.2**	**74.5**

(c) Bach10 dataset results for polyphony 3.

	Metric (%)			
Method	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	82.5	**86.1**	79.2	70.1
NBF	**83.6**	84.4	**82.9**	**71.8**

(d) Bach10 dataset results for polyphony 4.

Table 4.2: Evaluation of the Bach10 dataset before and after the refinement using neighbouring frames, for each polyphony level.

function $w_t(b)$ is applied to the pitch estimates, hence calculating each value of $W(b, q)$ as the weighted frequency of occurrence of a quantised $F0$ estimate. The refined polyphony estimate per frame is then used to select the pitches in $W(b, q)$ with highest histogram values.

The last step is to reconstruct the $F0$ values which have probably been correctly estimated. Firstly, one $F0$ candidate is created for each bin in the histogram $W(b, q)$. After that, for each bin, if an original $F0$ candidate for frame b happens to fall on that bin, it is likely that this pitch value is a true estimate, therefore the original value is used instead. If no original estimate for this frame b falls on, the value in $W(b, q)$ is then used for that bin. As in [17], the value used for R is 9, which corresponds to 90 ms for a 10-ms hop size.

Tables 4.2, 4.3, and 4.4 compares the results before and after the neighbouring frames refinement for the Bach10, MIREX and TRIOS datasets, respectively. For the Bach10 and MIREX collections, the evaluation is also performed for all polyphony combinations. The first row of the tables reveals the scores obtained via the NQ-Kraft algorithm, whereas the second row shows the results after the refinement, labelled by the acronym NBF (which stands for neighbouring frames).

It can be seen in the three tables that the system NQ-Kraft integrated with the

Method	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	**81.0**	**72.4**	89.0	**68.7**
NBF	80.1	68.4	**93.1**	67.1

(a) MIREX dataset results for polyphony 1.

Method	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	75.9	**73.7**	76.0	61.0
NBF	**76.8**	70.2	**82.2**	**61.8**

(b) MIREX dataset results for polyphony 2.

Method	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	71.8	**73.9**	67.3	55.5
NBF	**74.0**	70.7	**74.5**	**57.9**

(c) MIREX dataset results for polyphony 3.

Method	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	70.0	**72.5**	65.3	53.4
NBF	**73.1**	69.5	**72.9**	**56.7**

(d) MIREX dataset results for polyphony 4.

Method	Metric (%)			
	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	64.2	**74.4**	57.1	49.8
NBF	**67.7**	71.3	**64.7**	**53.5**

(e) MIREX dataset results for polyphony 5.

Table 4.3: Evaluation of the MIREX dataset before and after the refinement using neighbouring frames, for each polyphony level.

post-processing algorithm using neighbouring frames achieves better scores overall. Interestingly, for the evaluations of Bach10 and MIREX datasets, which comprises different degrees of polyphony, the best improvements in the F-measure scores were achieved for the most complex signals, which were approximately 1.1% and 3.5% for the highest polyphonies in the Bach10 and MIREX collections, respectively. In fact, for the monophonic signals the performance of the system after the refinement was actually worse, with a decrease of approximately 0.4% and 0.9% in the F-measure scores for the signals with one degree of polyphony for Bach10 and MIREX datasets, respectively.

As seen in Table 4.4, the results for the TRIOS dataset were significantly better after the refinement, figuring an improvement of around 4% for the F-measure score. An interesting result noticed in all evaluations is a considerable increase in the recall scores, followed by a slight decrease in the precision score. This means that the

	Metric (%)			
Algorithm	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	56.5	**78.3**	43.3	38.5
NBF	**60.5**	77.6	**48.3**	**42.3**

Table 4.4: Evaluation of the TRIOS dataset before and after the refinement using neighbouring frames.

post-processing refinement using neighbouring frames performs better in terms of reconstruction of non-detected pitches, than in terms of removing false positives.

4.3 Note tracking refinement

As mentioned in Chapter 1, a usual stage in AMT that follows MPE is note tracking, which is the second post-processing refinement. Like the neighbouring frames refinement described in previous section, note tracking is also based on temporal continuity and assumes that musical notes are locally stable. The goal of note tracking algorithms is to estimate musical notes by connecting pitch candidates that are close in both time and frequency.

Different attempts for note tracking have been proposed in the literature using several methodologies, including hidden Markov models [78] and conditional random fields [37]. Here, a modification of the MQ algorithm [68], which was originally proposed to address the partial tracking task, is implemented.

In the algorithm, notes can emerge (onset), remain active within frames, or vanish (offset). A note emerges when a pitch estimate is not associated with any other existing track, remains active while it is associated with pitch estimates, and vanishes when it finds no compatible estimate to incorporate. Let us suppose that p pitches were estimated in the frame b, with their respective fundamental frequencies denoted by $f_1, f_2, ..., f_p$. In the next frame $b+1$, r pitches have been detected with fundamental frequencies $g_1, g_2, ..., g_r$. The note tracking algorithm is explained as follows.

1. For each pitch g_i of frame $b+1$ a search is realised in order to find a note j which

had remained active until the frame b, satisfying the condition $|f_j - g_i| < \Delta f_j$. Parameter Δf_j controls the maximum frequency variation from one frame to the next one; in this work, it is adjusted to approximately 3%, corresponding to an approximate MIDI number variation of 0.3.

2. If pitch estimate g_i finds a corresponding note in the previous frame that satisfies the condition described in step 1, it associates with this note, which remains active. However, it can happen that two or more candidates can request to be associated with the same specific note. In this case, the pitch estimate with the closest fundamental frequency g_i with respect to note f_j wins the dispute, and the remaining ones will search for another note.

3. When a note j is not associated with any pitch estimate satisfying the condition in step 1, it is not considered active any more, and it is labelled to indicate that the note is vanishing or sleeping. In order to fix note discontinuities, a note can be sleeping within 100 ms, and if it finds a pitch estimate to associate with while sleeping, it can be labelled back as an active note. Otherwise, the note is terminated.

Except for the first frame, where all pitch candidates invariably start new notes, these steps are realised in all frames, until all pitches are labelled, that is, are part of a note. Lastly, short notes whose duration is less than 200 ms are removed. Figure 4.1, adapted from [65], illustrates the note tracking algorithm. It can be seen that, from frame $b-1$ to b, note f_1 is associated with the candidate g_1; thus, it remain active in frame b. Note f_2, sleeping for some frames, was not found by any pitch estimate and can be finished if it does not find any pitch within 100 ms. Candidate g_2 of frame b found note f_4, but it lost the dispute with pitch candidate g_3; therefore, pitch g_2 initialises a new note f_6.

Tables 4.5, 4.6, and 4.7 compares the results before and after the note tracking refinement for the Bach10, MIREX and TRIOS datasets, respectively. For the Bach10 and MIREX collections, the evaluation is also performed for all polyphony

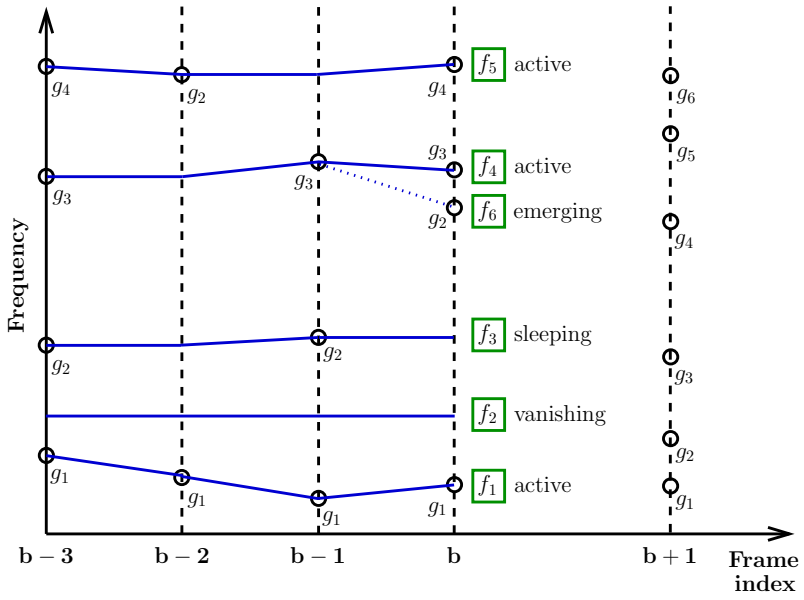

Figure 4.1: Scheme of the note tracking algorithm.

combinations. Like in the previous section, the first row of the tables reveals the scores obtained via the NQ-Kraft algorithm, whereas the second row shows the results after the refinement, labelled by the acronym NTR (which stands for note tracking).

It can be seen in Table 4.5 that, for the Bach10 collection, the NQ-Kraft method integrated with the post-processing algorithm using note tracking achieves better scores overall, for all degrees of polyphony. However, for the MIREX dataset, the second refinement only improved the results for the less complex signals, as seen in Table 4.6 for polyphony levels 1 and 2. For the full polyphony signals of MIREX dataset, the performance of the second refinement was approximately 1.3% worse than that of NQ-Kraft for the F-measure score. As seen in Table 4.7, the results for the TRIOS dataset were slightly better after the second refinement, figuring an improvement of around 0.3% for the F-measure score.

It can be concluded that the note tracking refinement performs differently for each dataset. The best improvements were obtained for the Bach10 dataset, followed by a slight improvement for the TRIOS collection. On the other hand, this refinement yielded worse scores for the more complex signals of the MIREX dataset.

	Metric (%)			
Method	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	91.3	88.6	**94.1**	84.0
NTR	**92.8**	**91.7**	94.0	**86.7**

(a) Bach10 dataset results for polyphony 1.

	Metric (%)			
Method	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	86.9	85.7	88.1	76.7
NTR	**88.2**	**88.0**	**88.4**	**78.9**

(b) Bach10 dataset results for polyphony 2.

	Metric (%)			
Method	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	84.8	85.7	83.9	73.4
NTR	**86.2**	**87.7**	**84.9**	**75.7**

(c) Bach10 dataset results for polyphony 3.

	Metric (%)			
Method	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	82.5	86.1	79.2	70.1
NTR	**84.0**	**87.9**	**80.5**	**72.4**

(d) Bach10 dataset results for polyphony 4.

Table 4.5: Evaluation of the Bach10 dataset before and after the refinement using note tracking, for each polyphony level.

	Metric (%)			
Method	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	81.0	72.4	**89.0**	68.7
NTR	**83.0**	**80.3**	83.6	**71.9**

(a) MIREX dataset results for polyphony 1.

	Metric (%)			
Method	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	75.9	73.7	**76.0**	61.0
NTR	**76.8**	**79.2**	74.2	**61.8**

(b) MIREX dataset results for polyphony 2.

	Metric (%)			
Method	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	**71.8**	73.9	**67.3**	**55.5**
NTR	71.5	**76.7**	64.5	54.9

(c) MIREX dataset results for polyphony 3.

	Metric (%)			
Method	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	**70.0**	72.5	**65.3**	**53.4**
NTR	69.1	**77.5**	62.9	51.7

(d) MIREX dataset results for polyphony 4.

	Metric (%)			
Method	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	**64.2**	74.4	**57.1**	**49.8**
NTR	62.9	**77.3**	52.7	46.5

(e) MIREX dataset results for polyphony 5.

Table 4.6: Evaluation of the MIREX dataset before and after the refinement using note tracking, for each polyphony level.

	Metric (%)			
Algorithm	$\mathcal{F}$	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{A}$
NQ-Kraft	56.5	78.3	**43.3**	38.5
NTR	**56.8**	**83.6**	42.2	**42.3**

Table 4.7: Evaluation of the TRIOS dataset before and after the refinement using note tracking.

4.4 Conclusion and final considerations

In this chapter, three different refinement algorithms were proposed to be integrated into the NQ-Kraft method, in order to improve the quality of the estimation of concurrent pitches. Firstly, it was proposed a user interaction approach, so that the degree of polyphony of the input signal can be informed to the system. Then a post-processing refinement from the literature that benefits from neighbouring frames was proposed to be integrated into the system. Lastly, a note tracking algorithm was proposed as a second refinement for the method.

Overall, the polyphony informed and neighbouring frames approaches yielded improvements in the scores, for all datasets. However, the note tracking refinement performed differently according to the dataset, specially for the MIREX collections, for which the refinement performed slight worse than the clean NQ-Kraft method. In the next chapter, conclusions and final considerations of this work are drawn.

Chapter 5

Conclusions and future work

This work focused on the implementation of a system for the automatic detection of multiple fundamental frequencies in polyphonic music signals. In Chapter 3, the reference method for multiple fundamental frequency estimation approached in this work was described, and a modification for it was proposed, referred to as NQ-Kraft, in order to allow non-integer MIDI values. Experimental results in three benchmark datasets show that the modified method outperforms its original version, and also figures among the best techniques when comparing it with state-of-the-art methods.

In Chapter 4, three refinement strategies were proposed to be incorporated into the NQ-Kraft method, in order to improve its performance. Firstly, a strategy allowing user interactivity was proposed to benefit from prior knowledge about the maximum polyphony number of the input signal. Following that, the second refinement strategy was a post-processing algorithm from the literature, which uses temporal information to remove wrong detections and reconstruct pitch estimates. Lastly, a note tracking algorithm was proposed in order to remove short duration notes as well as reconstruct discontinuities within notes.

Experimental tests reveal that the three proposed refinement algorithms improve the overall performance of the system. However, it is important to point out that each refinement strategy performs differently according to signal characteristics. While the first refinement achieved better results for less complex signals, the second one performed better for the most complex signals. As for the note tracking strategy,

it improved the results for the Bach10 and TRIOS datasets, whereas the performance of this strategy for MIREX dataset did not achieve better quality in multi-pitch estimation.

5.1 Future works

A careful reading of Chapter 3 reveals that the NQ-Kraft method is controlled by 15 parameters, and most of them can be adjusted only empirically. In fact, if the system is integrated with the refining algorithms, this number increases to 19. As a result, the realisation of experimental tests to calibrate them is considerably time consuming, and there are no guarantees that the optimal set of parameters has been reached. Therefore, a potential future step is to reduce the amount of free parameters in the system. A promising way to do that is to propose new models for isolated blocks of the method so that they no longer depend on so many parameters. For instance, the salience conditions may become obsolete with an improved version of salience functions.

The study of the computational complexity of the system was not addressed in this work, but a few comments about it are presented in the following. Experimental tests revealed that the system performs MPE at around 1 to 1.5 times the real duration of the input signals. It was also empirically perceived that the most complex stages of the method are the tonalness spectrum computation and the MCACF analysis. In order to reduce the processing time of the method, future works include the implementation of faster algorithms for these stages. For example, related works such as [17, 79] have employed simpler approaches for spectral peak selection, and in [33] the spectrotemporal representation is obtained via a 2-channel filterbank, which is less costly than the 5-channel filterbank employed in this work.

In this work, only three datasets were employed to assess the proposed NQ-Kraft method. In fact, these datasets concentrate many woodwind instruments, and this lack of generality can produce a biased system, specially for the proposed method, which requires a parameter tuning stage that terminates when overall performance

is considered satisfactory. Future steps of this work include the employment of more datasets suitable to MPE, such as the MAPS [51] and the Su [80] datasets.

The refining strategy allowing user interaction proposed in Section 4.1 is very straightforward. Its integration with the main system indeed yielded better estimation, however this direct approach can produce second or third harmonic errors, which are derived from strong pitch candidates. An improvement for this strategy could be to consider more sophisticated salience techniques. For example, Klapuri proposes in [3] an algorithm based on salience functions to deal with octave errors either jointly or iteratively.

In Section 4.2, it was proposed a refining algorithm that uses neighbouring frames, *i.e.* temporal information, to both remove incorrect detections and reconstruct non-detected true pitches. In this strategy, a weighted histogram is constructed by applying a triangular window to the pitch values of each semitone bin for each frame. This refinement could be potentially improved with a more sophisticated weighted histogram, which can be constructed using also magnitude or salience values of the pitch candidates.

Reference datasets usually comprise music signals with a high signal-to-noise ratio. However, music scenarios for AMT are diverse, and it is important to investigate the robustness of the system under different environments, such as smartphone playback or strong compression. One possible future work is to use the audio degradation toolbox [81], which applies different degradation classes into an input audio signals.

Bibliography

[1] KILMER, A. D. "The discovery of an ancient Mesopotamian theory of music", *Proceedings of the American Philosophical Society*, v. 115, n. 2, pp. 131149, Apr 1971.

[2] "The international society for music information retrieval". `http://www.ismir.net/`.

[3] Klapuri, A., Davy, M. (Eds.). *Signal processing methods for music transcription*. New York, Springer, 2006.

[4] BENETOS, E. *Automatic transcription of polyphonic music exploiting temporal evolution*. PhD Thesis, Queen Mary University of London, 2012.

[5] DUAN, Z., HAN, J., PARDO, B. "Multi-pitch streaming of harmonic sound mixtures", *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, v. 22, n. 1, pp. 138–150, Jan 2014.

[6] BENETOS, E., DIXON, S., GIANNOULIS, D., et al. "Automatic music transcription: challenges and future directions", *Journal of Intelligent Information Systems*, v. 41, n. 3, pp. 407–434, 2013.

[7] KLAPURI, A. *Signal processing methods for the automatic transcription of music*. PhD Thesis, Tampere University if Technology, Finland, 2004.

[8] DUAN, Z. *Computational music audio scene analysis*. PhD Thesis, Northwestern University, 2013.

[9] ŞENTÜRK, S., SERRA, X. "A method for structural analysis of Ottoman-Turkish Makam music scores". In: *6th International Workshop on Folk Music Analysis*, pp. 39–46, Dublin, Ireland, Jun 2016.

[10] FRIELER, K., PFLEIDERER, M., ZADDACH, W.-G., et al. "Midlevel analysis of monophonic jazz solos: a new approach to the study of improvisation", *Musicae Scientiae*, v. 20, n. 2, pp. 143–162, 2016.

[11] TZANETAKIS, G., COOK, P. "Musical genre classification of audio signals", *IEEE Transactions on Speech and Audio Processing*, v. 10, n. 5, pp. 293–302, Jul 2002.

[12] PRISCO, R. D., ESPOSITO, A., LETTIERI, N., et al. "Music plagiarism at a glance: metrics of similarity and visualizations". In: *21st International Conference Information Visualisation*, pp. 410–415, Jul 2017.

[13] DUDA, A., NÜRNBERGER, A., STOBER, S. "Towards query by singing/humming on audio databases". In: *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR)*, pp. 331–334, Vienna, Austria, Sep 2007.

[14] BENETOS, E., KLAPURI, A., DIXON, S. "Score-informed transcription for automatic piano tutoring". In: *Proceedings of the 20th European Signal Processing Conference (EUSIPCO'12)*, pp. 2153–2157, Aug 2012.

[15] RAPHAEL, C. "A Bayesian network for real-time musical accompaniment". In: *Advances in Neural Information Processing Systems (NIPS)*, 2001.

[16] Virtanen, T., Plumbley, M. D., Ellis, D. P. W. (Eds.). *Computational analysis of sound scenes and events*. Springer International Publishing, 2018.

[17] DUAN, Z., PARDO, B., ZHANG, C. "Multiple fundamental frequency estimation by modeling spectral peaks and non-peak regions", *IEEE Transactions on Audio, Speech, and Language Processing*, v. 18, n. 8, pp. 2121–2133, Nov 2010.

[18] PERTUSA, A. *Computationally efficient methods for polyphonic music transcription*. PhD Thesis, Universidad de Alicante, 2010.

[19] YEH, C. *Multiple fundamental frequency estimation of polyphonic recordings*. PhD Thesis, Université Paris VI, 2008.

[20] MÜLLER, M., ELLIS, D. P. W., KLAPURI, A., et al. "Signal processing for music analysis", *IEEE Journal of Selected Topics in Signal Processing*, v. 5, n. 6, pp. 1088–1110, Oct 2011.

[21] HARTMANN, W. M. "Pitch, periodicity, and auditory organization", *Journal of the Acoustical Society of America*, v. 100, n. 6, pp. 3491–3502, Oct 1996.

[22] MÜLLER, M. *Fundamentals of music processing: audio, analysis, algorithms, applications*. Springer, 2015.

[23] BELLO, J. P., DAUDET, L., ABDALLAH, S., et al. "A tutorial on onset detection in music signals", *IEEE Transactions on Speech and Audio Processing*, v. 13, n. 5, pp. 1035–1047, Sept 2005.

[24] YOUNG, R. W. "Terminology for logarithmic frequency units", *Journal of the Acoustical Society of America*, v. 11, n. 1, pp. 134139, Jul 1939.

[25] "Musical instrument digital interface". `https://www.midi.org/`.

[26] RAFFEL, C., ELLIS, D. P. W. "Extracting ground truth information from MIDI files: a MIDIfesto". In: *Proceedings of the 17th International Society for Music Information Retrieval Conference (ISMIR'16)*, New York City, USA, Aug 2016.

[27] EWERT, S., PARDO, B., MUELLER, M., et al. "Score-informed source separation for musical audio recordings: an overview", *IEEE Signal Processing Magazine*, v. 31, n. 3, pp. 116–124, May 2014.

[28] ORIO, N., LEMOUTON, S., SCHWARZ, D. "Score following: state of the art and new developments". In: *Proceedings of the 2003 Conference on New Interfaces for Musical Expression (NIME'03)*, pp. 36–41, Montreal, Canada, 2003.

[29] TOIVIAINEN, P., EEROLA, T. "MIDI toolbox 1.1". `https://github.com/miditoolbox/`, 2016.

[30] CHENG, T., DIXON, S., MAUCH, M. "A deterministic annealing EM algorithm for automatic music transcription". In: *Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR'13)*, Curitiba, Brazil, Nov 2013.

[31] SIGTIA, S., BENETOS, E., CHERLA, S., et al. "An RNN-based music language model for improving automatic music transcription". In: *Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR'14)*, Taipei, Taiwan, Oct 2014.

[32] BENETOS, E., WEYDE, T. "An efficient temporally-constrained probabilistic model for multiple-instrument music transcription". In: *Proceedings of the 16th International Society for Music Information Retrieval Conference (ISMIR'15)*, pp. 701–707, Málaga, Spain, Oct 2015.

[33] KRAFT, S., ZÖLZER, U. "Polyphonic pitch detection by iterative analysis of the autocorrelation function". In: *Proc. of the 17th International Conference on Digital Audio Effects (DAFx-14)*, Erlangen, Germany, Sep 2014.

[34] KRAFT, S., ZÖLZER, U. "Polyphonic pitch detection by matching spectral and autocorrelation peaks". In: *Proc. of the 23rd European Signal Processing Conference (EUSIPCO'15)*, pp. 1301–1305, Nice, France, Aug 2015.

[35] "Music information retrieval evaluation exchange (MIREX)". `http://www.music-ir.org/mirex/wiki/MIREX_HOME`.

[36] VINCENT, E., BERTIN, N., BADEAU, R. "Adaptive harmonic spectral decomposition for multiple pitch estimation", *IEEE Transactions on Audio, Speech, and Language Processing*, v. 18, n. 3, pp. 528–537, Mar 2010.

[37] BENETOS, E., DIXON, S. "Joint multi-pitch detection using harmonic envelope estimation for polyphonic music transcription", *IEEE Journal of Selected Topics in Signal Processing*, v. 5, n. 6, pp. 1111–1123, Oct 2011.

[38] BENETOS, E., DIXON, S. "Multiple-instrument polyphonic music transcription using a temporally constrained shift-invariant model", *Journal of the Acoustical Society of America*, v. 133, n. 3, pp. 1727–1741, Mar 2013.

[39] CARABIAS-ORTI, J. J., VIRTANEN, T., VERA-CANDEAS, P., et al. "Musical instrument sound multi-excitation model for non-negative spectrogram factorization", *IEEE Journal of Selected Topics in Signal Processing*, v. 5, n. 6, pp. 1144–1158, Oct 2011.

[40] GRINDLAY, G., ELLIS, D. P. W. "Transcribing multi-instrument polyphonic music with hierarchical eigeninstruments", *IEEE Journal of Selected Topics in Signal Processing*, v. 5, n. 6, pp. 1159–1169, Oct 2011.

[41] BENETOS, E., WEYDE, T. "Explicit duration hidden Markov models for multiple-instrument polyphonic music transcription". In: *Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR'13)*, Curitiba, Brazil, Nov 2013.

[42] FRITSCH, J. *High quality musical audio source separation.* Master's thesis, UPMC / IRCAM / Telécom ParisTech.

[43] BENETOS, E., CHERLA, S., WEYDE, T. "An efficient shift-invariant model for polyphonic music transcription". In: *Proceedings of the 6th International Workshop on Machine Learning and Music*, Prague, Czech Republic, Sep 2013.

[44] BAY, M., EHMANN, A. F., DOWNIE, J. S. "Evaluation of multiple-F0 estimation and tracking systems". In: *Proceedings of the 10th International*

Society for Music Information Retrieval Conference (ISMIR'09), Kobe, Japan, Oct 2009.

[45] DIXON, S. "On the computer recognition of solo piano music". In: *Australasian Computer Music Conference*, pp. 31–37, Brisbane, Australia, Jul 2000.

[46] DE CHEVEIGNÉ, A., KAWAHARA, H. "YIN, a fundamental frequency estimator for speech and music", *Journal of the Acoustical Society of America*, v. 111, n. 4, pp. 1917–1930, Apr 2002.

[47] MAUCH, M., DIXON, S. "PYIN: A fundamental frequency estimator using probabilistic threshold distributions". In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 659–663, May 2014.

[48] KLAPURI, A. P. "Multiple fundamental frequency estimation based on harmonicity and spectral smoothness", *IEEE Transactions on Speech and Audio Processing*, v. 11, n. 6, pp. 804–816, Nov 2003.

[49] KLAPURI, A. P. "A perceptually motivated multiple-F0 estimation method". In: *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 291–294, New Paltz, USA,, Oct 2005.

[50] KLAPURI, A. "Multipitch analysis of polyphonic music and speech signals using an auditory model", *IEEE Transactions on Audio, Speech, and Language Processing*, v. 16, n. 2, pp. 255–266, Feb 2008.

[51] EMIYA, V., BADEAU, R., DAVID, B. "Multipitch estimation of piano sounds using a new probabilistic spectral smoothness principle", *IEEE Transactions on Audio, Speech, and Language Processing*, v. 18, n. 6, pp. 1643–1654, Aug 2010.

[52] MEDDIS, R., O'MARD, L. "A unitary model of pitch perception", *Journal of the Acoustical Society of America*, v. 102, n. 3, pp. 1811–1820, Sep 1997.

[53] TOLONEN, T., KARJALAINEN, M. "A computationally efficient multipitch analysis model", *IEEE Transactions on Speech and Audio Processing*, v. 8, n. 6, pp. 708–716, Nov 2000.

[54] SMARAGDIS, P., BROWN, J. C. "Non-negative matrix factorization for polyphonic music transcription". In: *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 177–180, New Paltz, USA,, Oct 2003.

[55] CONT, A. "Realtime multiple pitch observation using sparse non-negative constraints". In: *Proceedings of the 7th International Society for Music Information Retrieval Conference (ISMIR'06)*, pp. 206–211, Victoria, Canada, Aug 2006.

[56] SMARAGDIS, P., RAJ, B., SHASHANKA, M. "A probabilistic latent variable model for acoustic modeling". In: *Workshop on Advances in Models for Acoustic Processing at NIPS*, Whistler, Canada, Dec 2006.

[57] BENETOS, E., DIXON, S. "A shift-invariant latent variable model for automatic music transcription", *Computer Music Journal*, v. 36, n. 4, pp. 81–94, Dec 2012.

[58] BENETOS, E., DIXON, S., GIANNOULIS, D., et al. "Automatic music transcription: breaking the glass ceiling". In: *Proceedings of the 13th International Society for Music Information Retrieval Conference (ISMIR'12)*, pp. 379–384, Porto, Portugal, Oct 2012.

[59] FLETCHER, N., ROSSING, T. *The physics of musical instruments*. 2nd edition ed. Berlin, Germany, Springer, 1998.

[60] DE CHEVEIGNÉ, A. "Multiple F0 estimation". In: Wang, D. L., Brown, G. J. (Eds.), *Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*, Wiley-IEEE Press, cap. 2, New York, 2006.

[61] HAYES, M. H. *Statistical digital signal processing and modeling*. 1st edition ed. New York, NY, USA, John Wiley & Sons, Inc., 1996.

[62] LICKLIDER, J. C. R. "A duplex theory of pitch perception", *Experientia*, v. 7, n. 4, pp. 128–133, 1951.

[63] MEDDIS, R., HEWITT, M. "Virtual pitch and phase sensitivity of a computer model of the auditory periphery — I: pitch perception", *Journal of the Acoustical Society of America*, v. 89, pp. 2866–2882, Jun 1991.

[64] KRAFT, S., LERCH, A., ZÖLZER, U. "The tonalness spectrum: feature-based estimation of tonal components". In: *Proc. of the 16th International Conference on Digital Audio Effects (DAFx-14)*, Maynooth, Ireland, Sep 2014.

[65] ESQUEF, P. A. A., BISCAINHO, L. W. P. "Spectral-based analysis and synthesis of audio signals". In: Pérez-Meana, H. M. (Ed.), *Advances in Audio and Speech Signal Processing: Technologies and Applications*, Idea Group, cap. 3, Hershey, USA, 2007.

[66] SMITH, J. O. *Spectral audio signal processing*. W3K Publishing, 2011.

[67] Zölzer, U. (Ed.). *DAFX: Digital audio effects*. 2 ed. New York, NY, USA, John Wiley & Sons, Inc., 2011.

[68] MCAULAY, R., QUATIERI, T. "Speech analysis/synthesis based on a sinusoidal representation", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, v. 34, n. 4, pp. 744–754, Aug 1986.

[69] SMITH, J., SERRA, X. "PARSHL an analysis/synthesis program for non-harmonic sounds based on a sinusoidal representation". In: *International Computer Music Conference*, pp. 290–297, Urbana, USA, Aug 1987.

[70] GEORGE, E. B., SMITH, M. J. "Analysis-by-synthesis/overlap-add sinusoidal modeling applied to the analysis and synthesis of musical tones", *J. Audio Eng. Soc*, v. 40, n. 6, pp. 497–516, 1992.

[71] NUNES, L. O., ESQUEF, P. A. A., BISCAINHO, L. W. P. "Evaluation of threshold-based algorithms for detection of spectral peaks in audio". In: *Proceedings of the 5th AES-Brazil Conference*, pp. 66–73, São Paulo, Feb 2007.

[72] BISHOP, C. M. *Pattern recognition and machine learning*. Secaucus, NJ, USA, Springer-Verlag New York, Inc., 2006.

[73] YOUNG, E. D., SACHS, M. B. "Representation of steady-state vowels in the temporal aspects of the discharge patterns of populations of auditory-nerve fibers", *Journal of the Acoustical Society of America*, v. 66, n. 5, pp. 1381–1403, Nov 1979.

[74] LAINE, U. K., KARJALAINEN, M., ALTOSAAR, T. "Warped linear prediction (WLP) in speech and audio processing". In: *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP-94)*, v. iii, pp. III/349–III/352 vol.3, Apr 1994.

[75] INDEFREY, H., HESS, W., SEESER, G. "Design and evaluation of double-transform pitch determination algorithms with nonlinear distortion in the frequency domain-preliminary results". In: *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP-85)*, v. 10, pp. 415–418, Apr 1985.

[76] PEETERS, G. "Music pitch representation by periodicity measures based on combined temporal and spectral representations". In: *IEEE International*

Conference on Acoustics Speech and Signal Processing Proceedings, v. 5, pp. V–V, May 2006.

[77] EMIYA, V., DAVID, B., BADEAU, R. "A parametric method for pitch estimation of piano tones". In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, v. 1, pp. I–249–I–252, April 2007.

[78] RYYNANEN, M. P., KLAPURI, A. "Polyphonic music transcription using note event modeling". In: *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics.*, pp. 319–322, New Paltz, USA,, Oct 2005.

[79] SERRA, X. *A system for sound analysis/transformation/synthesis based on a deterministic plus stochastic decomposition*. PhD Thesis, Stanford University, 1989.

[80] SU, L., YANG, Y.-H. "Escaping from the abyss of manual annotation: new methodology of building polyphonic datasets for automatic music transcription". In: *Int. Symp. Computer Music Multidisciplinary Research (CMMR)*, Jun 2015.

[81] MAUCH, M., EWERT, S. "The audio degradation toolbox and its application to robustness evaluation". In: *Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR'13)*, Curitiba, Brazil, Nov 2013.

[82] HARRIS, F. J. "On the use of windows for harmonic analysis with the discrete Fourier transform", *Proceedings of the IEEE*, v. 66, n. 1, pp. 51–83, Jan 1978.

Appendix A

Parabolic interpolation

The spectrum returned by the DFT is limited by a resolution of f_s/N_{DFT} Hz, where f_s and N represent respectively the sampling rate and the DFT length. Therefore, each frequency bin comprises an interval of f_s/N Hz, resulting in inaccuracy of peaks localisation for their both magnitude and frequency. Performing zero-padding in time domain increases the number of points in a same frequency range, thus increasing peaks localisation accuracy. However it is desirable a higher accuracy for the purpose of this work.

A very low computational cost and robust way for improving peak localisation is to perform a parabolic interpolation, since the surrounding region of a peak resembles a parabola [82]. This refinement for peak estimation was proposed by Serra & Smith ([69, 79]) in their work on sinusoidal analysis in the late 1980's. The method consists in fitting a parabola through the highest three samples of a peak (the peak itself and its two adjacent samples) and estimate the true peak magnitude and localisation in frequency, as illustrates the Figure A.1, adapted from [79].

Firstly, a coordinate system centred at the peak bin number $(k_p, 0)$ is defined. Using a general parabola expression

$$y(x) = a(x - p)^2 + b, \tag{A.1}$$

the objective is to estimate its centre p and height b, which are respectively the true

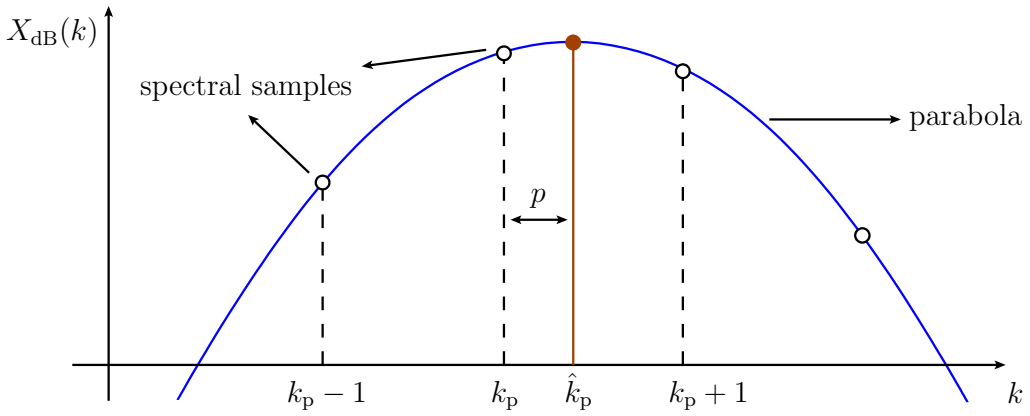

Figure A.1: Parabolic interpolation on a spectral peak. The empty circles represent the original samples in the spectrum whereas the solid one indicates the new localisation after the parabolic interpolation.

peak localisation and magnitude in the coordinate system aforementioned. Experimental results indicate that using magnitude scale in dB yields better accuracy, thus the values of the three highest samples are taken:

$$\begin{aligned} A_1 &\equiv X_{\mathrm{dB}}[k_p - 1] \\ A_2 &\equiv X_{\mathrm{dB}}[k_p] \\ A_3 &\equiv X_{\mathrm{dB}}[k_p + 1], \end{aligned} \tag{A.2}$$

where $X_{\mathrm{dB}}[k] = 20\log_{10}|X[k]|$. Solving the parabola equation, the centre p is

$$p = \frac{A_1 - A_3}{2(A_1 - 2A_2 + A_3)}, \tag{A.3}$$

the true peak location estimation in bins is

$$\hat{k}_\mathrm{p} \equiv k_\mathrm{p} + p, \tag{A.4}$$

and the true peak frequency is $\hat{k}_\mathrm{p} f_\mathrm{s}/N$. Finally, the true peak magnitude (in dB) can be also estimated:

$$X_{\hat{k}_\mathrm{p},\mathrm{dB}} = A_2 - \frac{p}{4}(A_1 - A_3). \tag{A.5}$$